

Udtrækning og Visualisering af Nøgledata fra Støjdata

Gregers W. Knudsen

Technical University of Denmark
Informatics and Mathematical Modeling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk

Summary

This thesis documents the development of an upgrade to an existing Brüel & Kjær Sound & Vibration A/S product, the BZ5503 – Utility Software for Hand-held Analyzers. The upgrade is developed in collaboration with the company and features a user interface upgrade, allowing easy recognition of measured data.

The project is done as a final exam project for the completion of a degree as certified engineer in information technology at the Technical University of Denmark.

The project was carried out during a period of 15 weeks, from 8th January 2007 to the 19th April 2007, with Stig Høgh from IMM as supervisor.

The upgrade to the BZ5503 user interface, gives access to a graphical overview of the measured data and key noise characteristics. As well as allowing for easy access to any metadata stored in the measurement.

Resume

Denne rapport dokumenterer udviklingen af en opgradering til et eksisterende Brüel & Kjør Sound & Vibration A/S produkt, BZ5503 – Utility Software for Hand-held Analyzers. Opgraderingen er udviklet i samarbejde med virksomheden og består af en opgradering af brugergrænsefladen, der tillader hurtig genkendelse af målte data.

Projektet er udarbejdet som afsluttende eksamensprojekt for Diplom-IT ingeniøruddannelsen ved Danmarks Tekniske Universitet (DTU).

Projektet er udført over en periode på 15 uger, fra d. 8. Januar 2007 til d. 19. April 2007, med Stig Høgh fra IMM som vejleder.

Opgraderingen til BZ5503s brugergrænseflade, giver brugeren adgang til et grafisk overblik over de målte data og nøgle støj karakteristika. Desuden giver den nem tilgang til eventuelle metadata, der er lagret sammen med målingen.

Vedlagt denne rapport

Der er til rapporten vedlagt en cd med screenshots af programmet og kildekode til de ændringer i BZ5503 der er foretaget.

På CD'en er der en folder kaldet "Kode", hvori der er placeret en tekst fil der indeholder referencer til alle filer der er oprettet eller ændret under projektet. Da implementeringen er foregået ind i eksisterende kode, er hele solutionen til BZ5503 vedlagt.

Tak Til

Jeg vil gerne takke Brüel & Kjør Sound &Vibration A/S for deres hjælp med tilblivelsen af dette projekt og for at stille udstyr, software mm. til rådighed.

Specielt vil jeg gerne takke hele udviklingsgruppen under Flemming Petersen, der med deres hjælpsomhed og ekspertise har støttet mig gennem hele forløbet.

Indhold

Summary	3
Resume	5
Vedlagt denne rapport	7
Tak Til	9
1. Projektdefinition	13
1.1. Problemstilling	13
1.1.1. Problemformulering	14
1.1.2. Projektafgrænsning	15
2. Definitioner	17
2.1.1. BZ5503 Screenshot	19
3. Kravspecifikation	21
3.1. Introduktion	21
3.1.1. Formål	21
3.1.2. Omfang	21
3.2. Overordnet beskrivelse	22
3.2.1. Produktperspektiv	22
3.2.2. Brugerkarakterstik	22
3.3. Specifikke krav	23
3.3.1. Automatisk visning/afspilning af annoteringer	23
3.3.2. Listning af relevante nøgledata for projekter	23
3.3.3. Grafisk visning af data indeholdt i projekter	24
3.3.4. Persistering af brugers kolonneopsætning	24
3.3.5. Fortsat integration af GridView i søgningsværktøjet	24
3.3.6. Erstatning af funktionalitet i nuværende ListView	25
3.3.7. Robust overfor fejl	25

4. Projektplanlægning	27
4.1. Risikoanalyse	28
4.1.1. Parallel udvikling.....	28
4.1.2. Brug af 3. parts komponent.....	28
4.1.3. Risiko for at GridView ikke kan tilpasses	29
4.2. Iterationsplan.....	30
4.2.1. Inception.....	30
4.2.2. Elaboration 1:	31
4.2.3. Elaboration 2:	31
4.2.4. Elaboration 3:	32
4.2.5. Elaboration 4:	32
5. Foranalyse.....	33
5.1. Teknologivalg	33
5.1.1. .NET 2.0 og C#.....	33
5.1.2. Developer Express – Extra Grid.....	33
5.2. Genbrug af GridView	34
6. Analyse.....	35
6.1. Nøgledata	35
6.1.1. Valg af data og visning	36
6.1.2. Tilvejebringelse af data.....	37
7. Design og implementering.....	39
7.1. Teknologianvendelse	40
7.1.1. Generics i .NET	40
7.2. Eksisterende struktur	41
7.2.1. ArchiveBrowser.....	42
7.2.2. TreeListView	43
7.2.3. SearchView	44
7.3. Opgradering af ArchiveBrowser	45
7.3.1. Erstatning af liste med GridView	46
7.3.2. Konsekvenser for GridView.....	47
7.3.3. Fortsat integration i SearchView	47
7.3.4. Persistering af kolonner	47
7.4. Udtræk og visning af basale nøgledata.....	48
7.5. Visning af grafiske data.....	50
7.6. Visning og afspilning af metadata.....	52
8. Test (Evaluering).....	53
8.1. Eksisterende B&K tests	54
8.2. Test af udvidet funktionalitet	55
8.3. Test med korruperede data.....	66
8.4. Performance evaluering	66

9. Udvidelsesmuligheder	67
9.1. Fejlrapportering ved korruperede data	67
9.2. Performance forbedringer	67
10. Konklusion.....	69
11. Kildehenvisninger	71
12. Bilag	73
12.1. Testrapport – Search.....	73
12.2. Testrapport – Archive Tasks	78

KAPITEL 1

1. Projektdefinition

1.1. Problemstilling

Brüel & Kjær Sound & Vibration A/S (B&K) udvikler og supporterer en række avancerede håndholdte lydanalysatorer, deriblandt den nyeste og hidtil mest avancerede model Type 2250. Denne lydanalysator kan lave en lang række forskellige målinger og databehandlinger, afhængigt af hvilket modul der aktiveres. Disse målinger kan derefter gemmes som projekter på apparatet, som siden hen overføres til et arkiv på en computer, ved hjælp af programmet BZ5503.

I analysatoren og arkivprogrammet, er der indbygget en række muligheder, for at tilføje metadata i form af kommentarer og annoteringer som lyd, tekst og billeder til sine projekter. Dette er gjort for, at man under og umiddelbart efter målingen kan beskrive omstændighederne for målingen eller specielle ting under målingen, så projektbeskrivelsen bliver så præcis som mulig. Dette blandt andet for, at muliggøre let identifikation af de enkelte projekter ved en senere rapportering. På trods af disse tiltag, erfarer B&K dog stadig at det er besværligt hurtigt at identificere enkelte målinger, blandt andet fordi disse oplysninger ikke er umiddelbart synlige i arkivprogrammet.

1.1.1. Problemformulering

Der skal designes og udvikles en udvidelse til arkivfunktionaliteten af BZ5503, for at muliggøre hurtig identifikation af projekter.

Løsningen skal fremhæve metadata samt karakteristiske måldata for den enkelte projekttype.

Det skal være muligt at:

- Automatisk vise og afspille annoteringer (metadata) til projekter og jobfoldere.
- Se relevante nøgledata for projekter direkte i listen.
- Vise en grafisk repræsentation af måldata direkte i listen, hvis projekttypen indeholder grafiske data.
- Persistere en brugers kolonneopsætning.
- Sikre fortsat integration af listen i søgeværktøjet (funktionalitet, som allerede findes i programmet).
- Erstatte den nuværende liste og dennes funktionalitet.
- Sikre robusthed overfor fejl og fejlbehæftede data.

Følgende skal undersøges i foranalysen:

- Hvilke data er umiddelbart tilgængelige og hvilke af disse kan betegnes som nøgledata?
- Er der forskel på hvilke data der er tilgængelige i projekterne, afhængigt af hvilken version af 2250 modulerne de er oprettet af?
- Er det muligt at bruge en af de i B&K eksisterende brugergrænseflader til listning af projekter?

1.1.2. Projektafgrænsning

Til projektet hører en række begrænsninger:

- Den nye måde at liste projekter på, ønskes kun implementeret i den del af BZ5503 der viser arkiverede projekter og ikke i den del der kan vise projekter på tilsluttede 2250ere. Kommunikationen mellem program og instrument fungerer i dag med en USB1.1 forbindelse, der ikke skønnes hurtig nok til at overføre alle de projektinformationer der vil blive nødvendige. Det formodes derfor, at udvidelsen ville blive betragtet mere som en gene end som end egentlig hjælp, hvis den blev bygget ind i denne del af programmet.
- Der skal kun bruges eksisterende data fra projekterne.
- Projekter fra 2250eren kan indeholde annoteringer, disse er basalt set bare en fil og kan derfor være af et vilkårligt format. Udvidelsen vil være begrænset til kun at håndtere lyd og tekst annoteringer, der er de mest almindelige typer.
- Udvidelsen skal kun supportere allerede udkomne projekttyper og de projekttyper der på nuværende tidspunkt er under udvikling, skal derfor ikke understøttes.

KAPITEL 2

2. Definitioner

2250: Type 2250 er en avanceret håndholdt lydanalysator, der er bygget op omkring en Windows CE kerne. Apparatet kan indeholde et antal software moduler, inklusiv frekvens analyse, logning (profilering) og optagelse af det målte signal. Disse målinger gemmes i projekter på 2250eren.

BZ5503: PC program der har redskaber til efterbehandlingsaktiviteter til Type 2250 lydanalysatoren. Programmet bruges til: synkronisering af opsætninger og data imellem en PC og en Type 2250, remote display af 2250, visning og arkivering af måledata og til dokumentation af måledata (eksport).

Nedenfor findes en gennemgang af BZ5503s brugergrænseflade (Se også figur 1):

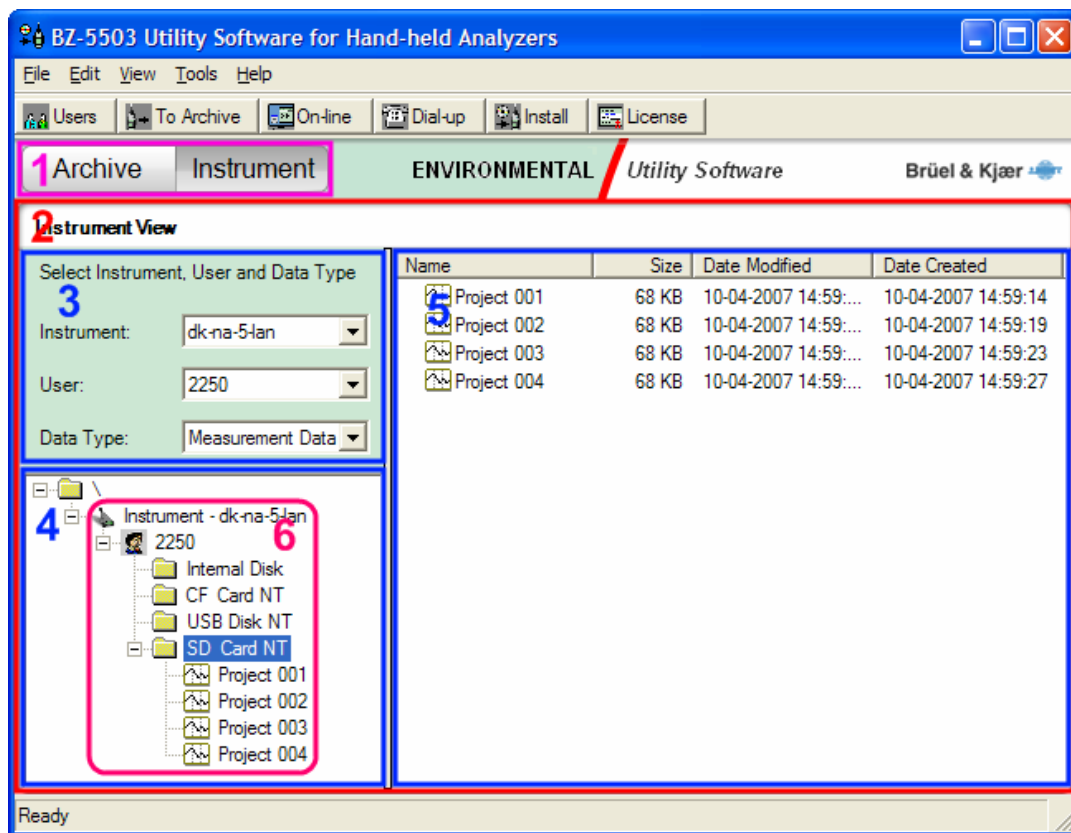
- **Task Selector:** Vælger hvilken opgavetype der skal være aktiv, altså om InstrumentBrowser eller ArchiveBrowser skal være aktiv.
- **InstrumentBrowser:** Brugergrænseflade til kommunikation med tilsluttede 2250ere. Overførsel af projekter til ArchiveBrowseren foregår herfra, ligesom der findes værktøjer til fjerstyring af 2250ere og lignende funktionalitet.
- **ArchiveBrowser:** Brugergrænseflade til arkivering af projekter, der også muliggør eksport og visning af projekter.
- **Filter:** Giver en valgmekanisme for hvad der vises i TreeView og ListView. Hvis en bruger har store mængder data, kan filteret hjælpe med at reducere datasættet.
- **TreeView:** Giver en mekanisme til at navigere igennem brugerens datasæt i InstrumentBrowseren eller ArchiveBrowseren.
- **ListView:** Viser detaljer for den i TreeView valgte ExplorerNode.
- **GridView:** Listview inspireret listning af ExplorerNoder.
- **SearchView:** Søgemechanisme udviklet under mit praktikophold, der kan søge rekursivt efter ExplorerNoder ud fra en start node.

ExplorerNode: Dette kan være alt lige fra et instrument tilkoblet til computeren, et arkiv, projekt eller lignende. Noderne er opbygget i en træstruktur, hvor der er en rod node øverst og hvor hver node refererer til sine undernoder. Alle noder under Rod niveauet repræsenterer en PersistentData struktur.

Nedenfor findes en gennemgang af de vigtigste node typer i ArchiveBrowseren:

- **Roden:** Den øverste node i hierarkiet og under hvilken alle andre noder findes. Under denne node kan der kun findes Arkiver.
- **Arkiv:** Data fra 2250eren overføres til et arkiv i BZ5503. Det er muligt at koble arkiver til og fra uden at de slettes på computeren, ligesom det er muligt at koble til et arkiv der ligger på en CD eller lignende. Under denne node kan der kun findes Brugere.
- **Bruger:** Indeholder to ting; 2250 opsætnings parametre og projekter, disse enten direkte på brugeren eller i en Job folder.
- **Job folder:** Struktur der bruges ligesom et bibliotek bruges i Microsoft Windows. Projekter der er optaget samme dag eller af samme bruger kan eksempelvis grupperes i disse. Job foldere kan dog også have Annoteringer under sig, hvilket kan bruges til eksempelvis at skrive noter til en række af projekter.
- **Projekt:** Er den egentlige måling foretaget med 2250eren. På nuværende tidspunkt eksisterer der syv projekttyper, der groft kan inddeles i tre hovedgrupper:
 - **Lydniveau måling:** En simpel måling af gennemsnitlig lydniveau over tid. Resultater i en række målte værdier (ca. 50), der hver i sær indeholder en enkelt dB værdier.
 - **Frekvensanalyse:** En måling der måler det maksimale, gennemsnitlige samt minimale lydniveau over tid, delt op i enten 1/1 oktav eller 1/3 oktav bånd. Resultatet er 5 spektre, der hver i sær indeholder en vektor med dB værdier.
 - **Logning:** En måling der gemmer (logger) ovenstående lydniveaumåling (eller dele heraf) med faste intervaller (f.eks. hvert sekund). Disse målinger kan også indeholde data fra ovenstående frekvensanalyse. Resultatet er en lang række målte parametre, der hver i sær indeholder en vektor af tidsperioder, der igen kan indeholde en vektor af lydniveauer i perioden, opdelt i et frekvensspektre.
- **Annotering:** Kan principielt være en hvilken som helst fil associeret med en Job folder eller et Projekt. Vil i de fleste tilfælde være en tekstfil eller en wav fil, skrevet eller optaget på 2250eren under målingen.

2.1.1. BZ5503 Screenshot



Figur 1: BZ5503 Screenshot

1. Task selector
2. InstrumentBrowser – ArchiveBrowser er lignende bygget op.
3. Filter
4. TreeView
5. ListView – Den der ønskes erstattet
6. ExplorerNoder

KAPITEL 3

3. Kravspecifikation

3.1. Introduktion

I samarbejde med B&K er der udviklet en kravspecifikation, med udgangspunkt i IEEE-830 standarden [IEEE-830]. Visse punkter er dog fjernet eller ændret af hensyn til deres relevans for dette projekt.

3.1.1. Formål

Denne specifikation beskriver kravene til en udvidelse til programmet BZ5503, der er et arkivprogram til lydanalytoren Type 2250.

Kravspecifikationen dokumenterer de krav der er aftalt i samarbejde med Brüel & Kjør Sound & Vibration Measurement A/S. Dokumentet danner derudover grundlag for designet af udvidelsen.

3.1.2. Omfang

Udvidelsen af BZ5503 skal muliggøre hurtig identifikation af arkiverede måleprojekter.

3.2. Overordnet beskrivelse

B&K havde selv en ide til hvordan listen skulle se ud. Listen skulle laves så den passede med det eksisterende ”look and feel” og der skulle helst ikke være nogen synlig forskel for brugeren, andet end ny funktionalitet.

3.2.1. Produktperspektiv

Udvidelsen skal fungere som en integreret del af programmet BZ5503 og erstatte den nuværende komponent til listning af projekter i en jobfolder og visningen af resultater af en søgning. Komponenten skal fuldt understøtte al nuværende funktionalitet, så som drag drop imellem komponenten og træ visningen.

3.2.2. Brugerkarakteristik

Brugerne af 2250 og BZ5503 er ofte konsulenter fra private virksomheder eller offentlige myndigheder, der typisk skal kontrollere om støj fra virksomheder giver anledning til, at lydniveauer i bebyggede områder overskrider nogle givne grænser. Grænseværdierne afhænger af ugedag og tidspunkt på døgnet. Visse typer støj (impulser og toner) giver straf tillæg.

Til en typisk vurdering, skal der ofte laves mange repræsentative målinger (forskellige aktivitetsniveauer på virksomheden i løbet af døgnet giver anledning til forskellige støjniveauer), som man efterfølgende skal vægte korrekt i forhold til hinanden i den endelige rapportering. Denne del lettes betydeligt, hvis man har tilknyttet kommentarer og noter direkte til de enkelte målinger, mens man foretog målingerne.

Brugerne bruger typisk BZ5503 til arkivering og eksport til andre B&K efterbehandlingsprogrammer, og på sigt vil BZ5503 også få efterbehandlings og rapporterings funktionalitet. Når data er arkiveret, har brugeren først og fremmest brug for at kunne genfinde og genkende dataene før en egentlig efterbehandling og rapportering kan finde sted.

Af hensyn til de knap 3000 brugere, der allerede bruger programmet, bør udvidelsen ikke ændre for meget på arbejdsgangen ved brug af programmet.

3.3. Specifikke krav

På baggrund af projektets problemformulering og afgrænsning, er der formuleret en række krav den endelige implementation skal opfylde. Der er lagt vægt på, at kravene er udformet og formuleret således, at det er muligt entydigt at teste om implementationen opfylder disse.

3.3.1. Automatisk visning/afspilning af annoteringer

Det skal være muligt, uden at klikke med musen, at få vist tekstannotering og afspillet lydannoteringer der er vedhæftet et projekt. Visningen skal foregå indenfor en rimelig tid og skal straks stoppe når brugeren skifter sin fokus væk fra projektet.

3.3.2. Listning af relevante nøgledata for projekter

Data der kan betegnes som nøgledata og muliggør hurtig identifikation af et projekt skal vises i kolonner udfor projektet. Det skal være muligt at sortere listen af projekter ud fra de værdier der måtte findes i disse kolonner.

Der skal laves en analyse, der afslører hvilke parametre der kan betegnes som nøgledata, for de enkelte projekttyper. Denne analyse skal laves i samarbejde med interne B&K brugerrepræsentanter og applikationsspecialister.

3.3.3. Grafisk visning af data indeholdt i projekter

I de projekter der indeholder loggede data eller et spektrum, skal data vises grafisk i en kolonne på linie med nøgledata for projektet. Brugeren skal yderligere kunne få et større vindue med det grafiske overblik, uden at skulle klikke med musen og uden at skulle åbne projektet for egentlig visning af data. Det forstørrede overblik skal forsvinde med det samme når brugeren fjerner sit fokus fra projektet.

Analysen der skal finde de vigtigste nøgledata for de forskellige projekttyper, skal samtidig bestemme hvilken parameter der skal benyttes til at lave det grafiske overview. Ligeledes skal en analyse afsløre, om der er forskel på hvilke data der er tilgængelige i projekter fra forskellige versioner af 2250 softwaren.

3.3.4. Persistering af brugers kolonneopsætning

Det skal være muligt for en bruger at udforme opsætningen af kolonner til dennes typiske opgaver. Kolonneopsætningen skal derfor være konfigurerbar og skal persisteres, også når programmet lukkes.

ArchiveExploreren og SearchView skal have hver deres opsætning.

I analysen af hvilke data der er nøgledata, skal det samtidig bestemmes hvilke kolonner der vil være synlige som default.

3.3.5. Fortsat integration af GridView i søgningsværktøjet

Selvom GridView udvides med avanceret funktionalitet, skal dennes fortsatte integration i Searchview sikres. Der skal ikke efter implementationen være ændrede arbejdsgange i SearchView.

SearchView må ikke have en forringet performance.

3.3.6. Erstatning af funktionalitet i nuværende ListView

Al nuværende funktionalitet i ArchiveBrowserens ListView skal bistå når GridView erstatter denne. Der må ikke være ændrede arbejdsgange ved brug af ArchiveExploreren efter implementationen.

ArchiveExploreren må ikke have en forringet performance.

3.3.7. Robust overfor fejl

Implementationen skal være robust og bør aldrig forårsage at BZ5503 crasher grundet problemer med et projekt, arkiv eller lignende.

Dette inkluderer, men er ikke begrænset til, robusthed overfor:

- Fejl på projektniveau, eksempelvis manglende filer.
- Manglende parametre i et projekts data.
- Korrumperede data i projekter.

KAPITEL 4

4. Projektplanlægning

Jeg har i tidligere projekter haft stor succes med at anvende Unified Process (UP), som udviklingsmetode for større IT-systemer og har derfor også brugt det i dette projekt.

Valget af Unified Process (UP) er baseret på nedenstående betragtninger:

- Iterativ og inkrementiel udvikling, hvilket giver mulighed for at vurdere systemet i helhed og foretage eventuelle tilpasninger af designet, efter hver iteration.
- Bruger Objekt Orienteret Analyse og Design (OOA/D).
- UML som notationsform giver standardiserede figurer som er lette at læse.
- Kontinuerlig test og revision.
- Nemt for B&K at følge udviklingen, da der er fungerende kode minimum efter hver iteration.

Før den egentlige planlægning af iterationer har jeg valgt at lave en risikoanalyse, da jeg på forhånd kunne se en række punkter der kunne forsinke projektet.

4.1. Risikoanalyse

4.1.1. Parallel udvikling

For at tilgå de nøgledata der er gemt i selve målingerne, vil det være nødvendigt at benytte en datastruktur, der på nuværende tidspunkt er på et udviklingsstadium, IMeasurement. Udviklingen foregår primært i en anden afdeling af B&K, men der er i projektgruppen for BZ5503 én person der er ansvarlig for tilpasningen til vores behov.

Problem:

Da IMeasurement stadig er på koncept stadiet, er det ikke muligt at sige hvorvidt dette projekt kan støde på større komplikationer i forløbet. Desuden udvikles konceptet til hele virksomheden og projektgruppens indflydelse på hvor mange ressourcer der vil blive brugt og præcis hvordan det vil se ud, er derfor begrænsede.

Forebyggelse:

For at minimere risikoen for at forsinkelser i udviklingen af IMeasurement skal blive et problem under projektet, lægges den del af implementationen der er afhængig af denne senest muligt i forløbet.

4.1.2. Brug af 3. parts komponent

GridView er i dag bygget op omkring et Extra Grid [DXG] fra Developer Express Inc.

Problem:

Jeg har ikke før min praktikperiode stiftet bekendtskab med Developer Express eller deres komponenter. Efter at have brugt komponenten, har det været min erfaring, at der er en del mindre fejl, der kan gøre det meget besværligt at arbejde med det.

Forebyggelse:

Ved at tage kontakt til en af de udviklere i B&K der har udviklet en anden komponent der benytter Extra Grid, håber jeg at kunne udnytte dennes erfaringer med produktet.

4.1.3. Risiko for at GridView ikke kan tilpasses

GridView er opbygget efter et internt B&K pattern der er inspireret af det alment kendte Model-View-Controller pattern.

Hele BZ5503 er opbygget efter et Model-View pattern.

Problem:

Hvis GridView skal erstatte det nuværende ListView vil det muligvis bryde med eller ikke passe ind i BZ5503s Model-View pattern.

Forebyggelse:

For at klarlægge tidligst muligt hvorvidt dette bliver et problem, vil dette blive undersøgt allerede i forundersøgelsen. I tilfælde af at GridView ikke kan tilpasses, vil der derfor være længst muligt tid til at udvikle en ny komponent, der er kompatibel med både SearchView og ArchiveBrowseren.

4.2. Iterationsplan

På baggrund af valget af Unified Process som udviklingsmetode, har projektperioden været opdelt i række iterationer.

Den første periode Inception, bestod hovedsageligt af foranalyse, udarbejdelse af kravspecifikation og udarbejdelse af rapportskabelon.

Herefter fulgte fire perioder med Elaborations, hvor den egentlige udvikling har foregået. I valget af rækkefølge af de fire Elaborations, er der taget hensyn til de, i risikoanalysen, opstillede problemstillinger.

Sidst i projektet, har der været en periode med afsluttende dokumentation og rapportskrivning.

Herunder ses en plan over de punkter der skal være dækket i hver elaboration. Planen er tilpasset løbende gennem projektet, efterhånden som jeg har lært mere om de enkelte punkter.

En iteration er først blevet beskrevet i detaljer, umiddelbart før afslutningen af den forrige iteration.

4.2.1. Inception

Foranalyse:

- Nøgledata
 - Hvilke data er nøgledata?
 - Hvordan tilgås de udvalgte nøgledata?
 - Hvordan skal nøgledata vises til brugeren?
- Kan GridView benyttes som erstatning til ListView?

Rapport:

- Udarbejd en rapportskabelon, ud fra det format der er påkrævet eksamensprojekter [SKA].

4.2.2. Elaboration 1:

GridView erstatter ListView i ArchiveBrowseren:

- Design hvordan GridView kan passes ind.
- Samarbejde mellem GridView og TreeView som der tidligere eksisterede mellem ListView og TreeView.
- Drag/Drop internt i GridView og til og fra TreeView.
- Koordination med ExplorerModelen.

Fortsat integration af GridView i SearchView:

- Sikre at generel listefunktionalitet virker.
- Sikre at Drag/Drop ikke er muligt.
- Sikre at Open/Export/Locate virker.

Persistering af kolonner i GridView:

- Persistering af brugers kolonneopsætning i GridView
- Individuel opsætning for SearchView og ArchiveBrowser

4.2.3. Elaboration 2:

Udtræk af nøgledata:

- Adgang til nøgledata.
- Oprettelse af struktur til lagring af nøgledata, da ExplorerModelen ikke understøtter disse ikke generelle data.

Visning af simple nøgledata i GridView:

- Oprettelse af kolonner til nøgledata.
- Visning af ikoner ved metadata på projekter

4.2.4. Elaboration 3:

Visning af grafik i GridView:

- Tegning af mini logging overview.
- Tegning af mini frekvens spektre

Popup funktionalitet i GridView:

- Annoteringer
 - Tilgang til annoteringer via PersistentData.
 - Visning af tekstannoteringer i tooltip.
 - Baggrundsafspilning af lydannoteringer.
- Overview graf
 - Brug af GCX til visning af logging profiler og frekvens spektre.

4.2.5. Elaboration 4:

Udførsel af funktionalitetstests op imod kravspecifikationen:

- Brug af gammel B&K test til sikring af at ArchiveExplorer og SearchView stadig virker normalt.
- Test af fortsat integration af GridView i SearchView udfra ny testplan.
- Test af ny GridView funktionalitet udfra nye testplaner.
- Test af GridViews robusthed overfor korruperede projekter og data udfra ny testplan.

5. Foranalyse

5.1. Teknologivalg

5.1.1. .NET 2.0 og C#

Udviklingen af BZ5503 startede i .NET 1.1 frameworket, men efter at NET 2.0 er kommet ud, er programmet porteret til dette framework. Da min løsning skal udvikles som en udvidelse til den eksisterende kode, har jeg valgt også at benytte mig af denne teknologi.

BZ5503 solutionen indeholder kode i forskellige sprog, men er primært udviklet i C#, hvilket også er grunden til at jeg har valgt dette sprog.

5.1.2. Developer Express – Extra Grid

Developer Express udvikler en række værktøjer til at lette arbejdet med grids, lister og lignende. I B&K benyttes Extra Grid allerede i en anden applikation til en kanalliste, hvor der bl.a. er realtime visning af input af de enkelte kanaler.

Da dette på mange måder ligner det der er ønsket i denne løsning, har jeg valgt at benytte Extra Grid til at opbygge GridView omkring.

5.2. Genbrug af GridView

I min praktikperiode udviklede jeg et søgningsværktøj til BZ5503, til udsøgning af projekter, annotering og lignende. Dette værktøj bestod af et todelt modulært vindue, med søgekriterier i toppen og en liste til udsøgte projekter nederst.

Da problemformuleringen til mit projekt allerede på dette tidspunkt var lavet, vidste jeg, at der formentligt ville blive brug for en mere fleksibel liste til listning af projekter end den daværende. Derfor udviklede jeg på det tidspunkt listen til søgevinduet som en selvstændig komponent, der dermed kunne genbruges og udvides under projektet.

Komponenten blev udviklet med et XtraGrid fra Developer Express Inc. som brugerflade. Denne komponent er meget konfigurerbar og blev allerede brugt til listning af andre ting, så som inputkanaler med realtime visning af inputtet, i en anden afdeling af B&K. Dette sikrede samtidig, at jeg kunne bibeholde B&K's "look and feel" i den nye komponent.

Listen blev udviklet på et tidspunkt, hvor jeg ingen erfaring havde med Explorer modellen og jeg kunne derfor ikke tage højde for den senere integration med denne under udviklingen af liste komponenten. Udviklingen af listen var med udgangspunkt i et internt B&K pattern, der er inspireret af det kendte Model-View-Controller pattern.

Det har dog siden hen vist sig at designet ikke passer særligt godt ind i Explorer modellen, da den forudsætter selv at holde alt data og vide alt om dets "view". Problemet er, at ExplorerModellen ikke understøtter at kunne indeholde de nye avancerede datatyper og den funktionalitet der nu bliver nødvendigt til den nye liste. ExplorerModellen er desuden en generel B&K komponent og kan derfor ikke udvides med BZ5503 specifik funktionalitet. Det bliver derfor nødvendigt at lave en del krumspring, for at kunne integrere den nye liste i den eksisterende brugerflade.

Konklusionen er dog, at jeg i projektet vil forsøge at tilpasse GridView i stedet for at udvikle en helt ny listefunktionalitet på ny. Da den forbedrede listefunktionalitet både er ønsket i SearchView og ArchiveBrowseren, betyder dette også at der kun skal udvikles og senere vedligeholdes kode et sted.

KAPITEL 6

6. Analyse

I analysefasen, af hver iteration, har jeg behandlet den funktionalitet der skulle udvikles eller revideres for at tilfredsstille de krav der var stillet i henhold til iterationsplanen.

Da det fra projektets start ikke var defineret helt præcis hvad konceptet nøgledata indeholdt, var det specielt i inception fasen nødvendigt med en stor del analyse.

Analysen skulle kortlægge en række punkter:

- Hvilke data ville hjælpe en bruger med nemt og hurtigt at genkende projekter fra hinanden og dermed kunne betegnes som nøgledata?
- Hvordan skulle disse nøgledata præsenteres for brugeren for at give det bedst mulige overblik?
- Hvor var disse udvalgte data gemt i projekterne?

6.1. Nøgledata

Til alle projekter på Type 2250 er der tilknyttet en applikationsspecialist, der har til opgave at have et stort kendskab til hvad lydmåleren rent faktisk bruges til. Analysen af hvilke data der er nøgledata og hvordan de skal vises er lavet i tæt samarbejde med denne applikationsspecialist og min projektleder, som også har stor erfaring indenfor området.

Applikationsspecialisten har en stor viden omkring lydmåleren og hvilke data der er tilgængelige, men ikke nødvendigvis hvordan det rent teknisk er lavet. Det er sparsomt med dokumentation omkring hvilke data der er tilgængelige og hvordan de tilgås i projekterne. Til at løse opgaven med at finde de relevante data i projekterne, har jeg måttet alliere mig med min projektleder og personer fra både BZ5503 og Type 2250 gruppen.

6.1.1. Valg af data og visning

Valget af hvilke data der kunne betragtes som nøgledata for en måling, tog udgangspunkt i hvilke data en bruger ville ønske. Dernæst blev der set på helt præcis hvilke data der var tilgængelige.

Efter de første møder med applikationsspecialisten og min projektleder, blev der formuleret en række ønsker til hvad en bruger ville ønske at kunne se ud fra listen:

- Hvilken lydmåler der havde været brugt til målingen.
- Indeholder projektet annoteringer og hvilken slags.
- Hvad er det egentlige start og stop tidspunkt for målingen.
- Hvor lang tid varede målingen.
- Data til at adskille målinger fra hinanden ud fra lydniveauet.
- Overblik-profil/spektre direkte i listen.
- I SearchView kan man have en interesse i at se stien til der hvor ens projekt er gemt.

Af de ønskede informationer var det kun det grafiske overblik der skilte sig ud. Alle andre informationer ville give et udmærket overblik ved blot at blive vist som tekst i en celle, hvilket også betyder at man ville kunne sortere tabellen efter disse.

Til det grafiske overblik direkte i listen, ønskede man en kraftigt nedsamlet version af de data der rent faktisk var indeholdt i projektet. Alle lognings eller frekvensanalyse projekter indeholder mindst én parameter der kan udtrykkes grafisk.

Det er dog forskelligt hvilken en parameter der er tale om, men der er i alle projekter enten en LAeq eller en LZeq. Disse parametre fortæller noget om det gennemsnitlige lydniveau indenfor en periode og vægtet med enten et A eller Z filter.

Da disse data altid er tilgængelige og giver et godt overblik over dataene, blev den af disse der eksisterer i den enkelte måling valgt til overblikket.

Kravspecifikationen beskriver også, at der skal kunne ses et mere detaljeret overblik over de grafiske data i projekter der indeholder dette. For at lette tilgangen til dette, vil det blive lavet som et tooltip, der vises når brugeren holder musen over kolonnen med den lille grafiske representation.

Ud over de informationer der var ønskelige i listen, var der, som det er udtrykt i kravspecifikationen, en interesse i at gøre annoteringer hurtigt tilgængelige. Det var her yderst vigtigt for applikationsspecialisten, at det ikke måtte kræve andet interaktion fra brugeren end at flytte musen, for at se disse. Det blev derfor bestemt at tekstannoteringer skulle vises som et tooltip, altså blot ved at holde musen over det ikon i listen, der fortalte om der var nogle tekstannoteringer på projektet. På samme måde skulle lydannoteringer begynde at afspille blot ved at brugeren holder musen over ikonet for disse, på projektet.

6.1.2. Tilvejebringelse af data

Med en beskrivelse af, hvilke data der kunne betegnes som nøgledata i hånden, kunne arbejdet med at tilvejebringe disse data begynde.

Det stod fra starten klart, at der skulle hentes data fra et antal forskellige steder, og at de sværeste data at tilgå formentligt ville blive de egentlige måledata.

Nedenfor er en liste over de steder der senere skulle hentes data ud fra og hvilke data der hentes:

- **ExplorerNode:** Struktur oprettet af ArchiveBrowser, der indeholder generelle informationer omkring noden, bruges til at bygge træstrukturen i TreeView.
 - Hvilke annoteringer der findes til projekter og jobfoldere.
 - Stien til noden.
- **PersistentData:** Struktur oprettet på 2250eren der indeholder informationer om noder, bl.a. størrelse og den direkte sti til filen på disken.
 - Data indeholdt i annoteringer, bruges til visning og afspilning af disse.
- **IAnnotation:** XML fil gemt i roden af alle projektfoldere på disken. Indeholder informationer som bruges internt af 2250eren og BZ5503.
 - Indeholder serienummer på den 2250er der er brugt under målingen.

De sidste informationer det var ønsket at synliggøre for brugeren var alle gemt i de egentlige måledata. Disse data er lagret binært i et internt B&K format kaldet en "recording", og har hidtil ikke kunnet tilgås fra BZ5503. Dette har man dog tidligere kommet uden om, ved at åbne en NT version af 2250 modellen og vist data i et vindue, der svarer til det man ville se på 2250eren.

Samtidig med mit projekt blev der i BZ5503 gruppen arbejdet på en DataViewer, der skulle erstatte 2250 vinduet til visning af projektdata. Til denne opgave hørte også, at kunne trække data ud fra det format som 2250 dataene er lagret på disken. Dette ville også muliggøre, at jeg ved hjælp af det nye format, der ville blive oprettet i forbindelse med denne opgave, havde adgang til måledata.

I dataene fra 2250eren ligger der bl.a. en "total"-måling, der indeholder de fleste af de informationer der er nødvendige, for at skabe det grafiske billede af logget data, der er ønsket i GridView. Det eneste problem er, at denne kun i de aller nyeste versioner af 2250 SW'eren der indeholder "total."-målinger for loggede data.

Den eneste mulighed for, at skabe disse data, udover at læse dem direkte i total målingen, er at nedsample de egentlige data. Dette er dog en større og tidskrævende process, og det blev besluttet at det ikke ville være muligt at vise disse overviews for gamle projekter med loggede data.

- **IMeasurement:** Nyt dataformat der udvikles sideløbende med mit projekt. Dataformatet indeholder alle rå data fra målingerne og alle deskriptorer der beskriver disse.
 - Egentlige start/stop tider for målingerne.
 - Gennemsnitligt, maksimum og minimum lydniveau for måleperioden.
 - Data til generering af grafisk repræsentation af målingen.

KAPITEL 7

7. Design og implementering

Da projektet er udviklet efter Unified Process metoden, har jeg arbejdet iterativt og derfor er designet og koden også udviklet iterativt.

Der har efter hver iteration været et fungerende produkt, der derefter kunne videreudvikles i den næste iteration. En iteration bestod af helt ny funktionalitet eller omskrivelse og tilpasning af det eksisterende.

Det, at der fandtes en fungerende løsning, som minimum efter hver iteration, gjorde det muligt løbende at se om det der var blevet implementeret fungerede i praksis. Samtidig blev det lettere for både min projektleder og applikationsspecialister hos B&K at følge udviklingen. Input fra disse kilder, har igennem forløbet været uvurderligt, blandt andet fordi disse besidder en stor viden om hvordan vores kunder rent faktisk bruger produktet.

Yderligere er debugmulighederne større, når koden kan eksekveres.

Jeg ønskede, at have noget kode der så tidligt som muligt i en iteration kunne eksekveres og give en idé om det endelige resultat. Dette opnåedes ved blandt andet at bruge dummy data, indtil der var adgang til de virkelige data.

Udviklingsafdelingen jeg har siddet i på B&K, arbejder alle i den samme kodebase. Dette betyder blandt andet, at flere personer ofte arbejder i de samme Visual Studio solutions og i nogle tilfælde også i de samme filer.

For at styre tilgangen til de forskellige filer og til at holde en versionshistorik over de forskellige filer bruges der programmet Subversion, der er et opensource program.

De følgende afsnit beskriver de overvejelser, jeg har gjort mig i forbindelse med designet og implementationen af løsningen.

7.1. Teknologianvendelse

I projektet har jeg benyttet en række teknologier der er tilgængelige i .Net frameworket.

7.1.1. Generics i .NET

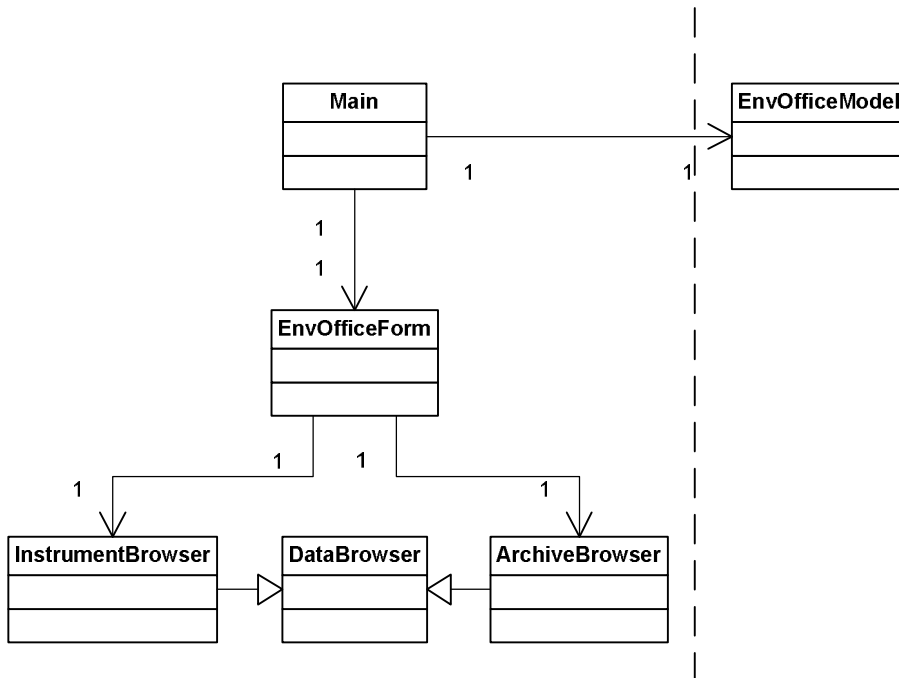
I .NET 2.0 er der kommet en række såkaldte generics [GEN], heriblandt List<T>. Generics konceptet går kort fortalt ud på, at man kan erklære en generic med en vilkårlig type, hvorefter den vil være bundet til denne type, de er med andre ord typestærke efter oprettelse.

Det kan sammenlignes med et array, hvor hvis man opretter et string-array kan man kun ”putte” strings i det.

Et eksempel på en generic, som vi har benyttet meget i dette projekt er List. List minder meget om et array, men har bl.a. den fordel at den kan gennemløbes med f.eks. en foreach statement og at den fungerer som en collection, hvilket er en stor fordel når elementer ofte skal tilføjes og fjernes fra Listen.

7.2. Eksisterende struktur

BZ5503 er bygget op omkring et ModelView pattern, hvor klassen Main er indgangen til hele PC softwaren. Denne instancierer EnvOfficeModel og EnvOfficeForm, der reagerer på alle bruger input og kommunikerer med modellen:

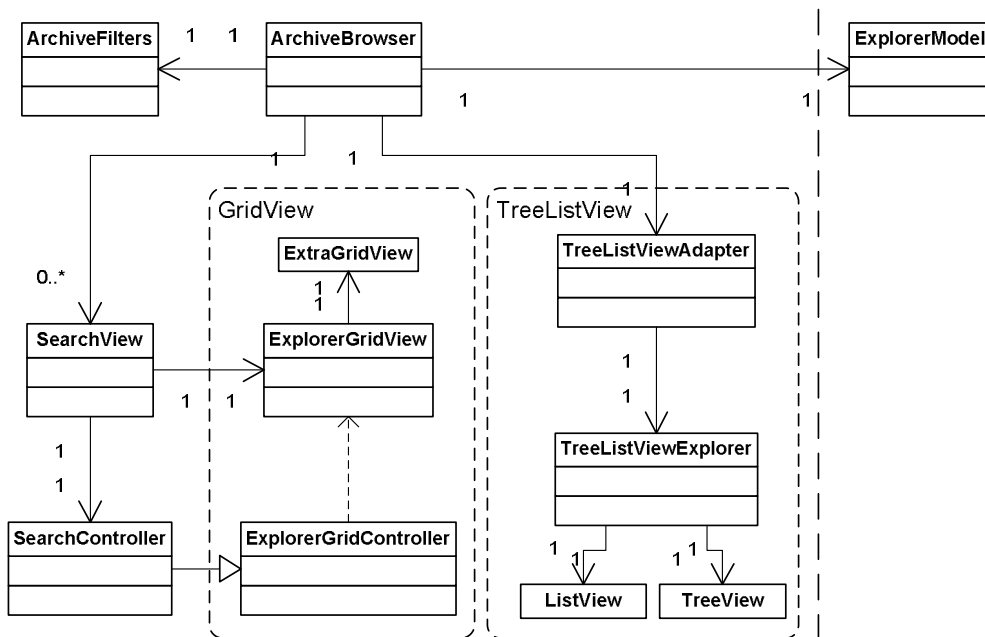


Figur 2: Generel BZ5503 struktur

EnvOfficeForm indeholder de to views der findes i programmet, henholdsvis Instrument- og Archive-Browser. Da disse har meget tilfælles og begge benytter ExplorerNode strukturen til at opbygge et træ af noder, er denne funktionalitet samlet i DataBrowser.

7.2.1. ArchiveBrowser

Hoved brugergrænsefladen i ArchiveBrowseren består af et filter, et træ og en liste. TreeListView ses nedenfor som den komponent der indeholder træet og liste og sørger for samarbejdet mellem disse. Dette view holdes opdateret gennem events fra ExplorerModellen. Filteret begrænser her hvilke noder der er tilgængelige i TreeListView.



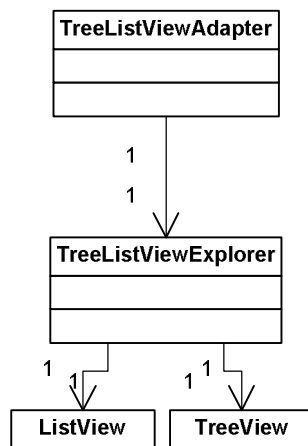
Figur 3: ArchiveBrowser opbygning, før projektstart

I min praktikperiode tilføjede jeg endnu et view. SearchView er et modulært view der viser et fastfrosset billede af hvilke noder der kunne findes ved en rekursiv søgning, startende fra en startnode. SearchView opdateres altså ikke af events fra ExplorerModellen, men får stadig sit indhold derfra.

7.2.2. TreeListView

TreeListView adapter fungerer i dette view som en controller, der koordinerer de to views, henholdsvis ListView og TreeView. Det er også her at alle metoder til opfyldning af træ og liste findes, sammen med alle kommandoer til ExplorerModellen.

Kopiering, åbning af noder og alle andre kommandoer sendes herfra til ExplorerModellen, der derfra finder ud af hvilken handling der skal ske.



Figur 4: TreeListView

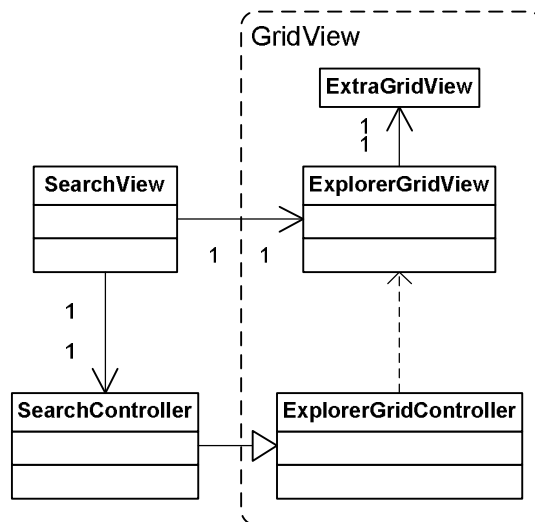
Når der skal udvikles et lignende TreeGridView, vil det være nødvendigt at lave en sammenlignelig struktur, da ArchiveBrowseren forventer at dets view følger et IexplorerViewAdapter interface.

7.2.3. SearchView

Det eksisterende SearchView består af en Form, SearchView klassen, og en controller, der foretager selve udsøgningen af noderne. Til visning af nodelisten er der her forsøgt lavet et generelt komponent der kunne genbruges i mit projekt.

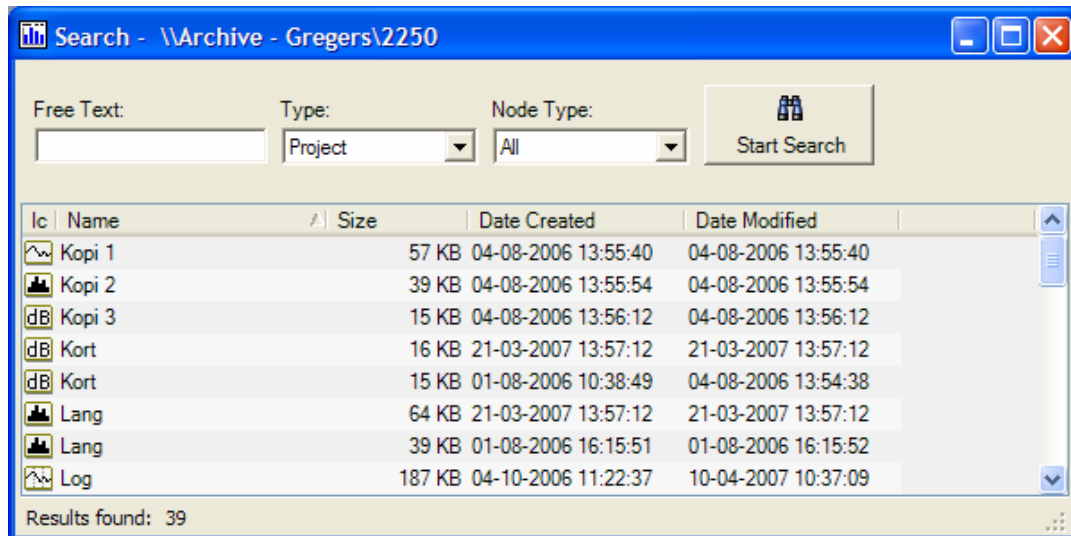
Komponenten GridView består af et view og en controller. Viewet har en ExtraGrid fra Developer Express og håndterer opsætningen og alle events fra denne. Alle kommandoer der kræver at ExplorerModellen eller ArchiveBrowseren skal hjælpe, eksempelvis ved åbning af en node, bliver eksekveret ved kald ned i controlleren.

ExtraGridet i ExplorerGridView er databundet med en datatabel i ExplorerGridControlleren, der indeholder alle metoder til tilføjelse og fjernelse af noder fra tabellen. Controlleren indeholdt ligeledes alle metoder til kommunikation til ExplorerModellen og ArchiveBrowseren.



Figur 5: SearchView, før projektstart

Nedenfor ses et screenshot af Search, hvor kun de mest basale informationer om de udsøgte noder er tilgængelige.



Figur 6: Search, før projektstart

7.3. Opgradering af ArchiveBrowser

For at kunne opgradere ArchiveBrowseren med det nye GridView, var der en del opgaver der skulle løses.

Først og fremmest skulle der laves en TreeListView lignende struktur, en TreeGridView, der kunne erstatte den nuværende. Dette krævede en del ændringer, både i den adapter der tidligere styrede samarbejdet med de to views, men mest af alt ændringer i GridView.

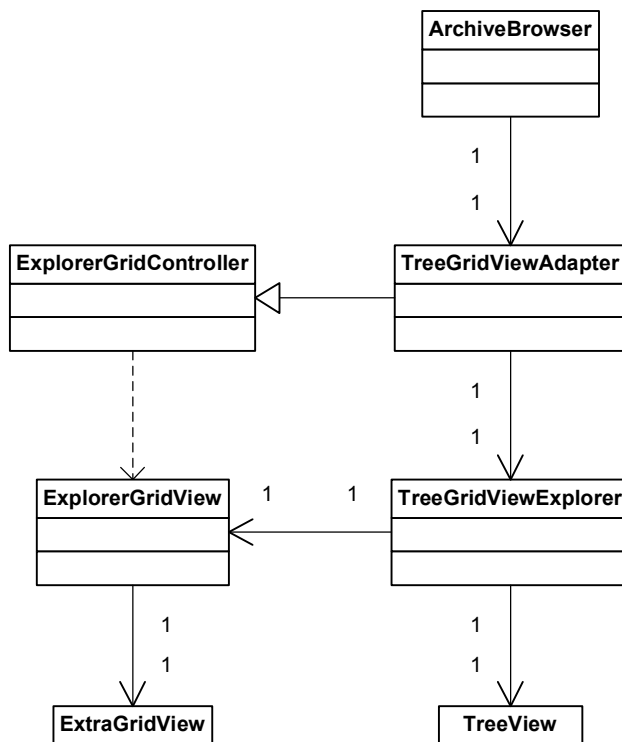
Konsekvenserne for GridView kunne derefter kortlægges og dens fortsatte integration i SearchView sikres.

Sidst var der en opgave der var lidt uafhængig af de andre opgaver, men dog vigtig, nemlig den ønskede persistering af en brugers kolonneopsætning.

7.3.1. Erstatning af liste med GridView

For at erstatte brugen af en ListView i TreeListView med en GridView, har jeg valgt at følge samme struktur som til SearchView. Adapteren fungerede i forvejen som en slags controller for de to views, der sørgede for at holde træet og listen opdateret og håndterede al kommunikation med ExplorerModellen.

SearchView, som GridView tidligere var lavet til, havde dog ikke brug for den samme intensive kommunikation med ExplorerModellen som det nu er nødvendigt. Da ExplorerModellen forventer, at få en event for alt hvad der sker i og omkring de to views (lige fra når musen kommer ind i et view, til den står over en node eller når en node forsøges åbnet), skulle der foretages nogle ændringer i GridView.



Figur 7: TreeGridView

7.3.2. Konsekvenser for GridView

ExplorerGridView rapporterer nu alle events videre til ExplorerControlleren, der indeholder en række virtuelle metoder. Disse metoder kan overskrives af enten TreeGridAdapteren eller af SearchControlleren, der begge nedarver fra denne.

I de virtuelle metoder, er der allerede skrevet kode til rapportering til ExplorerModellen.

I GridView har det også været nødvendigt, at tilføje mulighed for multiselect og drag/drop, hvilket har krævet noget implementation. Desuden skal den kontekstmenu der frembringes ved højreklik på musen nu komme fra ExplorerModellen, der holder styr på hvilke handlinger der er mulige på en node eller en samling af disse.

7.3.3. Fortsat integration i SearchView

Det har været nødvendigt at implementere en ExplorerModel lignende minimodel i SearchControlleren, da denne nu blandt andet forventes, at vide hvilken node der er valgt.

De virtuelle metoder der kommunikerer med ExplorerModellen er her overskrevet, så de i stedet kommunikerer med denne minimodel, eller i nogle tilfælde slet ikke gør noget.

7.3.4. Persistering af kolonner

Det var oprindeligt ønsket at kolonneopsætningen skulle persisteres, hver gang der blev lavet en ændring i denne. Dette var dog desværre ikke muligt, da Extra Grid ikke kan sende en event når dette sker.

Derfor er kommandovejen til persistering af kolonnerne lavet så SearchView eller applikationen fortæller, at nu lukkes det aktuelle vindue og listen af kolonner der er åbne og deres bredde skal gemmes.

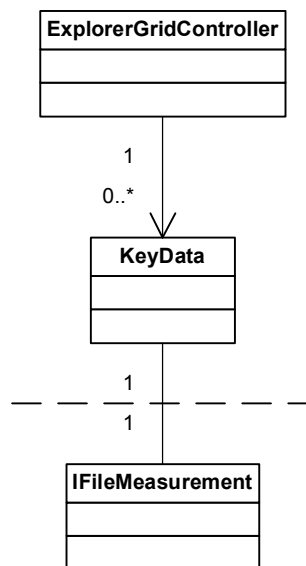
Disse informationer bliver derefter hentet og gemt i en XML fil i AllUsers på maskinen. Samme sted som alt andet information i øjeblikket gemmes af BZ5503.

7.4. Udtræk og visning af basale nøgledata

Nogle af de ønskede nøgledata var umiddelbart tilgængelige på den enkelte ExplorerNode, så som stien til nodens placering. Disse informationer bliver hentet af ExplorerGridControlleren og skrevet ind i kolonnerne ligesom alle andre informationer.

Til at hente de data der lå gemt i IAnnotation og IFilemeasurement blev der oprettet en klasse der håndterer udtræk og opbevaring af disse.

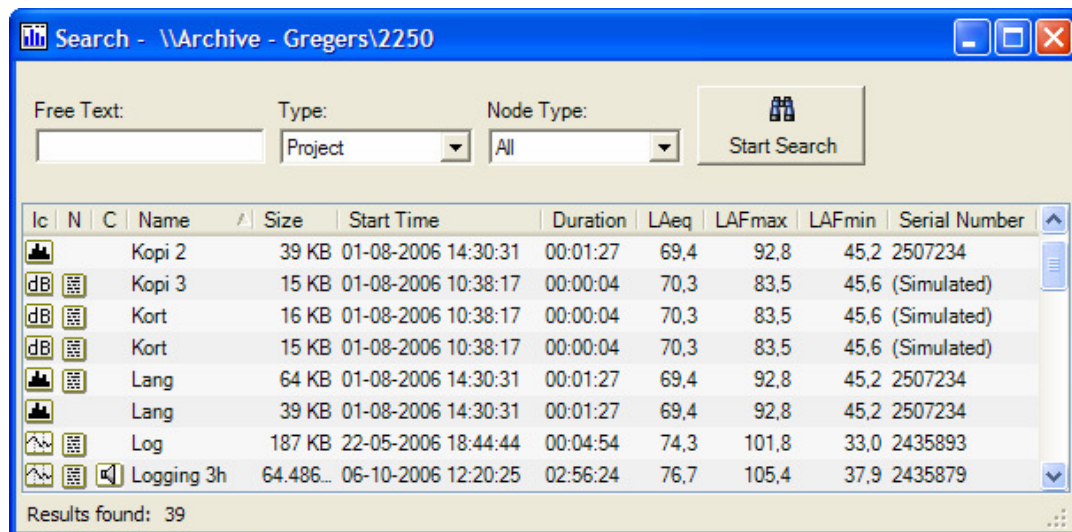
IFilemeasurement er et helt nyt og meget generelt dataformat der stadig er på udviklingsstadiet. Dataformatet er lavet så generelt, for at alle måledata data som enhver B&K måler eller applikation genererer til enhver tid skal kunne gemmes heri.



Figur 8: ExplorerGridController med KeyData

IFilemeasurement er opbygget ved, at man har en række Funktioner der i princippet er deskriptorer der beskriver de data der er gemt i målingen. Data er gemt i en multibuffer, der under sig har en række sekvenser. Det er i disse sekvenser at den egentlige data ligger.

Efter at have trukket de basale nøgledata ud, endte det med en liste der kunne vise informationer som set nedenfor (Figur 9).

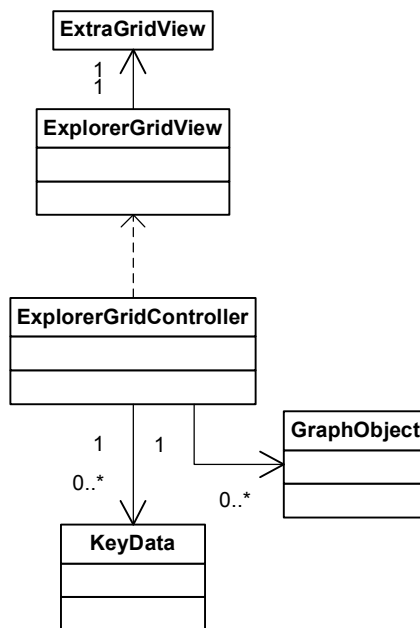


Ic	N	C	Name	Size	Start Time	Duration	LAeq	LAFmax	LAFmin	Serial Number
			Kopi 2	39 KB	01-08-2006 14:30:31	00:01:27	69,4	92,8	45,2	2507234
			Kopi 3	15 KB	01-08-2006 10:38:17	00:00:04	70,3	83,5	45,6	(Simulated)
			Kort	16 KB	01-08-2006 10:38:17	00:00:04	70,3	83,5	45,6	(Simulated)
			Kort	15 KB	01-08-2006 10:38:17	00:00:04	70,3	83,5	45,6	(Simulated)
			Lang	64 KB	01-08-2006 14:30:31	00:01:27	69,4	92,8	45,2	2507234
			Lang	39 KB	01-08-2006 14:30:31	00:01:27	69,4	92,8	45,2	2507234
			Log	187 KB	22-05-2006 18:44:44	00:04:54	74,3	101,8	33,0	2435893
			Logging 3h	64.486...	06-10-2006 12:20:25	02:56:24	76,7	105,4	37,9	2435879

Figur 9: SearchView med basale nøgledata

7.5. Visning af grafiske data

I KeyData udtrækkes der også data til at lave den grafiske repræsentation af målingen. For at sikre at der senere kan skrives et nyt view til controlleren, oprettes der i controlleren en klasse der skal håndtere tegningen af profilerne og spektrene.

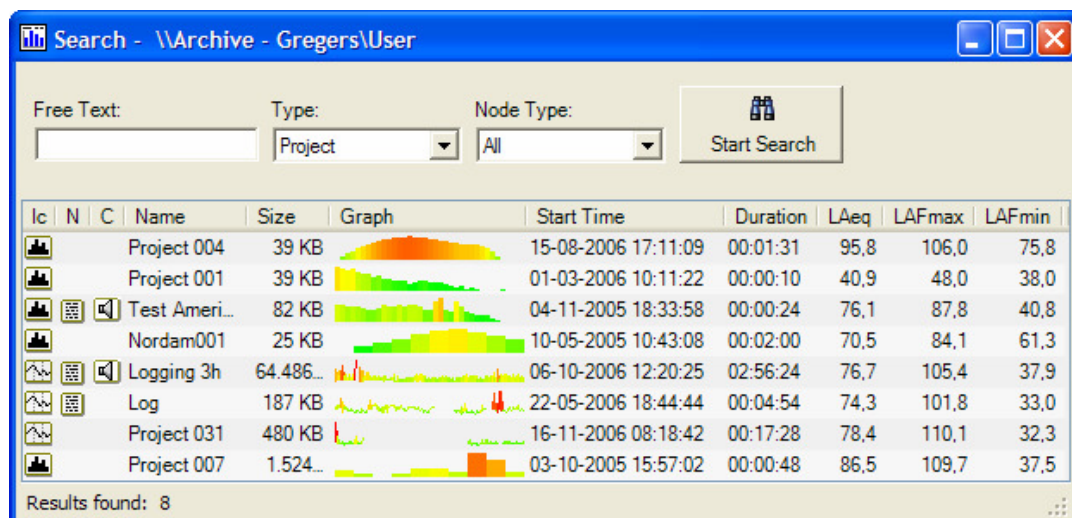


Figur 10: GridView, efter projekt

GraphObject placeres i en skjult kolonne i datatabellen og indeholder ud over de data der er nødvendige for at tegne, også en metode til dette. ExplorerGridView kan sende et grafik objekt og en ønsket størrelse af den ønskede grafiske overblik og skal derefter blot placere det i den rigtige celle.

Tegningen i ExplorerGridView sker i en event, der kan bruges til at tegne indholdet af en celle i stedet for at placere data i denne.

Med de nye grafiske data ser Search vinduet ud som nedenfor (Figur 11).



Figur 11: SearchView med grafisk data

På screenshottet er Graph kolonnen i farver og ikke sort-hvid som i det første udkast af denne iteration.

Forslaget med at farve indholdet i Graph kolonnen, kom efter at have holdt et revisionsmøde med den tilknyttede applikationsspecialist og min projektleder for iterationen.

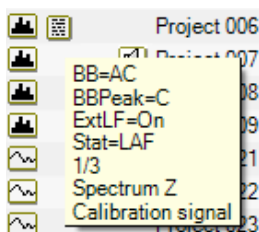
Indholdet af Graph kolonnen er farvet efter lydtrykket for det givne tidspunkt eller frekvens. Skalaen går fra helt grøn, ved lydtryk under 30dB, og helt rødt, ved lydtryk over 100dB.

Dette var ikke en del af den oprindelige kravspecifikation, men det viser en af styrkerne ved at udvikle UP principperne.

7.6. Visning og afspilning af metadata

Der findes i PersistentData metoder til at hente teksten ud af en tekstannotering. For at kunne vise flere tekstannoteringer i det samme tooltip, er der i controlleren lavet en metode der konkatenerer teksten fra en liste af tekstannoteringer til en samlet tekststreng. Denne streng kan derefter vises af ExplorerGridView i et standard tooltip.

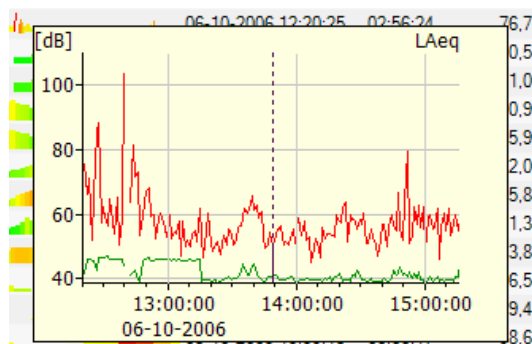
Afspilningen af lydannoteringer foregår i ExplorerGridView og det er kun den første lydannotering på noden der bliver afspillet.



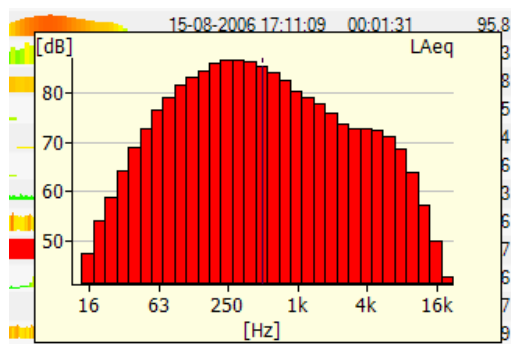
Figur 12: Tekst annotering tooltip

Der findes ikke i Extra Grid en event, der fortæller når musemarkøren kommer ind i en bestemt celle. For at lave et tooltip der indeholder grafik har det derfor været nødvendigt konstant at checke på MouseMove eventen om musemarkøren befinder sig over en celle indeholdende et GraphObject.

Selve tooltipet er lavet ved, at der åbnes en Form der ikke kan få fokus, hvorpå man tegner de ønskede data ind, ved hjælp af en B&K komponent. Denne komponent – GCX – er et grafikelement udviklet til at vise spektre og profiler.



Figur 13: Profil tooltip



Figur 14: Spektre tooltip

KAPITEL 8

8. Test (Evaluering)

Til projektet er der lavet en række funktionalitetstest, der sikrer at alle krav er opfyldt. Der er ikke i BZ5503 projektgruppen tradition for at lave egentlige unittest af koden, men det overvejes dog i øjeblikket hvordan sådanne kunne etableres.

For at teste op imod kravspecifikationen er der benyttet en række forskellige testtyper.

Der findes allerede testrapporter, der tester hvorledes træ og listefunktionaliteten i BZ5503 fungerer. Da et af kravene var at denne funktionalitet skulle være bibeholdt, nu hvor listen er erstattet med GridView, er denne test blevet brugt til at verificere dette. Ligeledes fandtes der allerede testplaner der beskrev hvorledes SearchViews eksisterende funktionalitet skulle testes.

Til at verificere, at den nye funktionalitet der er blevet tilføjet GridView virker, er der blevet udformet en række nye testplaner.

Der er blevet lavet en række forsøg, hvor GridView fodres med nogle projekter hvor der er manglende data eller fejl i disse. Dette for at sikre BZ5503 ikke crasher når det udsættes for et eventuelt korrump arkiv.

BZ5503 med den udvidede funktionalitet i GridView er sidst sendt ud til evaluering hos kunderepræsentanter. Det der ønskedes evalueret var hvorvidt kravene om ikke forringet performance var overholdt.

8.1. Eksisterende B&K tests

Der eksisterer i dag testrapporter til at teste den tidligere funktionalitet i ArchiveBrowser og SearchView. For at sikre at implementationen lever op til kravene 4 og 5 fra kravspecifikationen, er disse tests blevet kørt.

Kravene til implementationen var at den nye funktionalitet ikke måtte komme på bekostning af noget af den eksisterende funktionalitet.

Efter at have kørt testrapporterne igennem kan det konkluderes at den tidligere funktionalitet stadig er intakt. Testrapporterne er at finde vedlagt rapporten i appendiks 1 og 2.

8.2. Test af udvidet funktionalitet

Efterhånden som de forskellige delelementer af udvidelsen blev færdige, er der udført test af disse op imod kravspecifikationen.

Til hvert krav er der blevet udarbejdet én eller flere testplaner, der beskriver hvorledes testen skal udføres, samt hvilke forudsætninger der skal være opfyldt, for at testen kan udføres. Tillige er der forventede resultat beskrevet, samt nødvendige handlinger ved fundne fejl.

Følgende er en liste over de testplaner der er lavet:

Testplan 1 – Afspilning og visning af annoteringer

Testplan 2 – Tilgængelighed af relevante nøgledata

Testplan 3a – Visning af grafisk data overview i listen

Testplan 3b – Visning af grafisk data overview i tooltip

Testplan 4 – Persistering af kolonner

På de næste sider findes testplanerne og de tilhørende testrapporter. Der fandtes ingen fejl under den endelige testfase.

TESTPLAN

Krav 1 – Afspilning og visning af annoteringer

Testplan ID.:	version.:	Testtype:	Dato:	Tester:
1	1.0	Krav spec. test	10-04-07	GWK

Testformål:

Test at GridView kan vise indholdet af tekst annoteringer og afspille lyd annoteringer.

Testforudsætninger:

- En node (job eller bruger) er valgt
- Den valgte node indeholder projekter med tekst og lyd annoteringer.
- Mindst én node i listen indeholder mere end én tekstannotering

Udførsel:

1. Flyt musemarkøren hen over højttaler ikonet for et projekt indeholdende en lydannotering. Hold musen stille.
 - a. Lydannoteringen afspilles fra start til slut.
2. Flyt musemarkøren hen over højttaler ikonet for et projekt indeholdende en lydannotering. Flyt musen efter et kort øjeblik.
 - a. Lydannoteringen afspilles, men stoppe det øjeblik musemarkøren forlader ikonet.
3. Flyt musemarkøren hen over en node indeholdende mindst én tekstannotering.
 - a. Et tooltip med teksten fra annoteringen vises.
4. Flyt musemarkøren hen over en node indeholdende mindst to tekstannoteringer.
 - a. Et tooltip med den samlede tekst fra tekstannoteringerne vises.

Korrigerende handlinger ved fundne fejl:

Kildekoden rettes til og testen foretages igen.

TESTRAPPORT

Krav 1 – Afspilning og visning af annoteringer

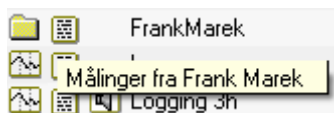
Testrapport ID: 1	version: 1.0	Testtype: Krav spec. test	Udført: 10-04-07	Tester: GWK
-----------------------------	------------------------	-------------------------------------	----------------------------	-----------------------

Testresultat:

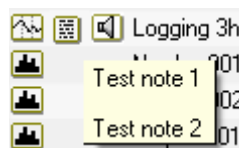
1 & 2:

Lydannoteringerne afspilles i begge tilfælde korrekt og stopper når musemarkøren forlader cellen.

3:



4:



Konklusion:

Testen forløb som forventet.

TESTPLAN

Krav 2 – Tilgængelighed af relevante nøgledata

Testplan ID.:	version.:	Testtype:	Dato:	Tester:
1	1.0	Krav spec. test	10-04-07	GWK

Testformål:

Test at GridView indeholder de relevante nøgledata for de enkelte projekter.

Testforudsætninger:

- En node (job eller bruger) er valgt
- Den valgte node indeholder projekter af alle tre hovedgrupper.

Udførsel:

1. Slå alle kolonner med nøgledata til i GridView.
 - a. Se at "Start Time", "Stop Time" og "Duration" er korrekt udfyldt for alle typer af projekter.
 - b. Se at "LAeq", "LAFmax", og "LAFmin" er korrekt udfyldt (B vejede projekter indeholder ikke data til disse kolonner).
 - c. Se at "Serial Number" er korrekt udfyldt for alle typer af projekter (Projekter genereret i simulatoren er mærket med "(Simuleret)" i denne kolonne).
 - d. Se at "Path" peger på den korrekte sti til projektet for alle projektyper.

Korrigerende handlinger ved fundne fejl:

Kildekoden rettes til og testen foretages igen.

TESTRAPPORT

Krav 2 – Tilgængelighed af relevante nøgledata

Testrapport ID: 1	version: 1.0	Testtype: Krav spec. test	Udført: 10-04-07	Tester: GWK
-----------------------------	------------------------	-------------------------------------	----------------------------	-----------------------

Testresultat:

Icon	Name	Start Time	Stop Time	Duration	LAeq	LAFmax	LAFmin	Serial Number	Path
	L2010	29-03-2007 13:27:28	29-03-2007 13:27:38	00:00:10	63,5	66,9	61,1	2449915	\\Archive - Gregers\2250\
	L2010	29-03-2007 12:31:32	29-03-2007 12:31:47	00:00:15	55,9	57,2	54,8	2449915	\\Archive - Gregers\2250\
	L2010	20-03-2007 15:19:13	20-03-2007 15:19:24	00:00:11	56,7	57,5	56,0	2506593	\\Archive - Gregers\2250\
	Log	22-05-2006 18:44:44	22-05-2006 18:49:38	00:04:54	74,3	101,8	33,0	2435893	\\Archive - Gregers\2250
	Logging 3h	06-10-2006 12:20:25	06-10-2006 15:16:49	02:56:24	76,7	105,4	37,9	2435879	\\Archive - Gregers\2250
	Nordam001	10-05-2005 10:43:08	10-05-2005 10:45:08	00:02:00	70,5	84,1	61,3	2479689	\\Archive - Gregers\2250
	Nordam002	10-05-2005 10:46:14	10-05-2005 10:49:14	00:03:00	71,0	79,1	67,7	2479689	\\Archive - Gregers\2250
	Project 001	01-03-2006 10:11:22	01-03-2006 10:11:32	00:00:10	40,9	48,0	38,0	2479716	\\Archive - Gregers\2250

Results found: 172

Konklusion:

Alle data er tilstede i listen og som forventede.

TESTPLAN

Krav 3a – Visning af grafisk data overview i listen

Testplan ID.: 1	version.: 1.0	Testtype: Krav spec. test	Dato: 10-04-07	Tester: GWK
---------------------------	-------------------------	-------------------------------------	--------------------------	-----------------------

Testformål:

Test at GridView kan vise en grafisk repræsentation af logge og spektre projekter, direkte i listen.

Testforudsætninger:

- En node (job eller bruger) er valgt
- Den valgte node indeholder projekter af logge og spektre hovedgrupperne.
- Loggeprojekterne er lavet med version 2.0 eller nyere af BasicEnv softwaren.

Udførsel:

1. Slå kolonnen "Graph" til i GridView.
 - a. Der vises, for loggede projekter, en min/max indhylningskurve der repræsenterer de egentlige loggede data.
 - b. Der vises, for spektre projekter, et søjlediagram med et antal søjler svarende til antallet af målte frekvensbånd.
 - c. Graferne er farvekodede efter lydniveauet ved det enkelte tidssample/frekvensbånd.

Korrigerende handlinger ved fundne fejl:

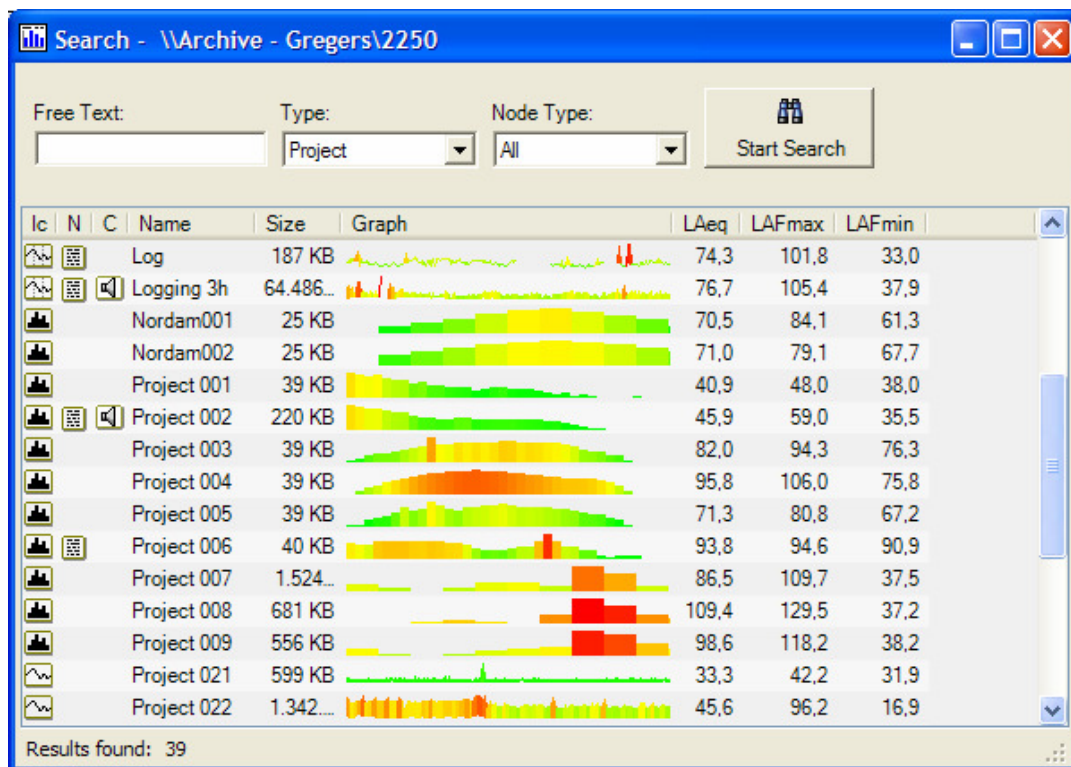
Kildekoden rettes til og testen foretages igen.

TESTRAPPORT

Krav 3a – Visning af grafisk data overview i listen

Testrapport ID: 1	version: 1.0	Testtype: Krav spec. test	Udført: 10-04-07	Tester: GWK
-----------------------------	------------------------	-------------------------------------	----------------------------	-----------------------

Testresultat:



Konklusion:

Der vises korrekte farvekodede grafer for begge typer projekter. Der fandtes ingen fejl.

TESTPLAN

Krav 3b – Visning af grafisk data overview i tooltip

Testplan ID.: 1	version.: 1.0	Testtype: Krav spec. test	Dato: 10-04-07	Tester: GWK
---------------------------	-------------------------	-------------------------------------	--------------------------	-----------------------

Testformål:

Test at GridView kan vise en grafisk repræsentation af logge og spektre projekter, i et tooltip.

Testforudsætninger:

- En node (job eller bruger) er valgt
- Den valgte node indeholder projekter af logge og spektre hovedgrupperne.
- Loggeprojekterne er lavet med version 2.0 eller nyere af BasicEnv softwaren.
- Kolonnen ”Graph” er slået til i GridView.

Udførsel:

1. Flyt musemarkøren hen over cellen med et grafisk overview af et projekt med loggede data.
 - a. Se at der kommer et tooltip frem, der viser en max og min kurve for loggeperioden.
 - b. Se at akserne har de rigtige enheder og skalerede korrekt.
 - c. Se at vægtningen i øverste højre hjørne, svarer overens med den vægtning der blev anvendt ved målingen.
2. Flyt musemarkøren hen over cellen med et grafisk overview af et projekt med spektre data.
 - a. Se at der kommer et tooltip frem, der viser et søjlediagram der repræsenterer det målte frekvens spektre.
 - b. Se at akserne har de rigtige enheder og skalerede korrekt.
 - c. Se at vægtningen i øverste højre hjørne, svarer overens med den vægtning der blev anvendt ved målingen.
3. Fjern musemarkøren fra en ”Graph” celle.
 - a. Se at tooltipet forsvinder med det samme.

Korrigerende handlinger ved fundne fejl:

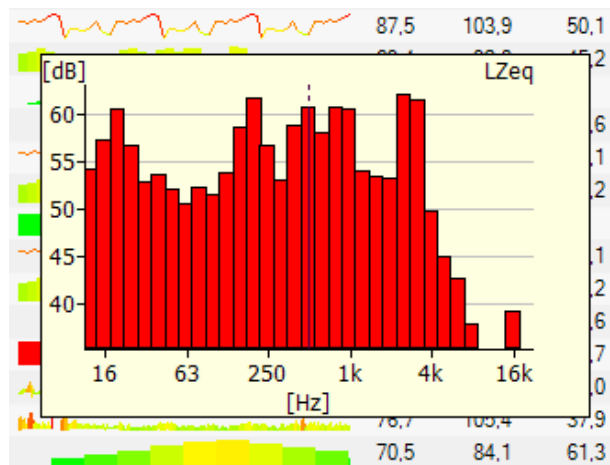
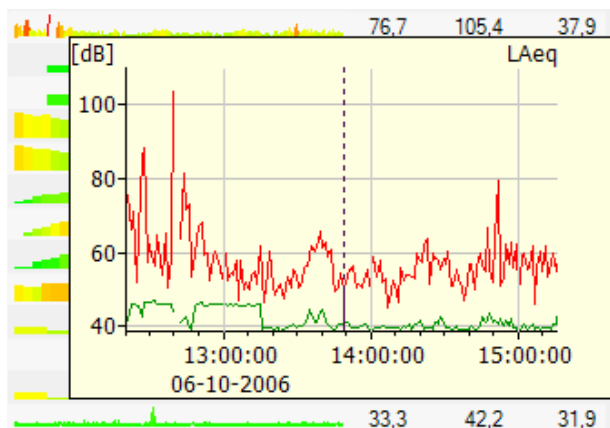
Kildekoden rettes til og testen foretages igen.

TESTRAPPORT

Krav 3b – Visning af grafisk data overview i tooltip

Testrapport ID: 1	version: 1.0	Testtype: Krav spec. test	Udført: 10-04-07	Tester: GWK
-----------------------------	------------------------	-------------------------------------	----------------------------	-----------------------

Testresultat:



Konklusion:

De to typer tooltips vises korrekt og forsvinder når musemarkøren fjernes fra cellen.

TESTPLAN

Krav 4 – Persistering af kolonner

Testplan ID.: 1	version.: 1.0	Testtype: Krav spec. test	Dato: 10-04-07	Tester: GWK
---------------------------	-------------------------	-------------------------------------	--------------------------	-----------------------

Testformål:

Test at kolonneopsætningen persisteres individuelt for Search og Arkiv browseren.

Testforudsætninger:

- BZ5503 er startet op
- Archive er valgt i task selectoren

Udførsel:

- | | |
|--|---|
| <ol style="list-style-type: none"> 1. Start Search og lav om i kolonneopsætningen, husk denne. 2. Luk Search. 3. Start Search <ol style="list-style-type: none"> a. Verificer at kolonneopsætningen ikke har ændret sig. 4. Lav igen om i kolonneopsætningen 5. Luk Search og luk BZ5503. 6. Start BZ5503. 7. Start Search <ol style="list-style-type: none"> a. Verificer at kolonneopsætningen ikke har ændret sig. | <ol style="list-style-type: none"> 1. Lav om i kolonneopsætningen i ArchiveBrowser. 2. Luk BZ5503 3. Start BZ5503 <ol style="list-style-type: none"> a. Verificer at kolonneopsætningen ikke har ændret sig. |
|--|---|

Korrigerende handlinger ved fundne fejl:

Kildekoden rettes til og testen foretages igen.

TESTRAPPORT

Krav 4 – Persistering af kolonner

Testrapport ID: 1	version: 1.0	Testtype: Krav spec. test	Udført: 10-04-07	Tester: GWK
-----------------------------	------------------------	-------------------------------------	----------------------------	-----------------------

Testresultat:

- Alle kolonneopsætninger persisteres efter lukning og genåbning af Search.
- Alle kolonneopsætningen persisteres i Search efter lukning og genåbning af BZ5503.
- Alle kolonneopsætninger persisteres i ArchiveBrowseren efter lukning og genåbning af BZ5503.

Konklusion:

Testen forløb som forventet og det fandtes ingen problemer.

8.3. Test med korruperede data

Robustheden af den færdige implementation er blevet testet ved at fjerne essentielle filer fra projekter og ved at forsøge at vise projekter med kendte fejl i dataene. Det er dog ikke lykkedes at få BZ5503 til at gå ned på grund af dette. Kravet om at programmet under ingen omstændigheder må crashe grundet korrupte projekter er dermed overholdt.

Selvom programmet ikke går ned, når der forsøges at vise projekter der er fejlbehæftede på den ene eller den anden måde, er der dog stadig plads til forbedringer. I flere tilfælde vil projekterne slet ikke dukke op på listen over projekter og brugeren må i dette tilfælde selv ud og finde dem på disken, for at kunne slette eller rette op på disse. En delvis løsning på dette problem er beskrevet i kapitel 9, Udvidelsesmuligheder.

8.4. Performance evaluering

For at evaluere hvorvidt kravet til performance er overholdt, er det færdige program blevet sendt ud til et antal interne kunderepræsentanter. Kravet var at udvidelsen ikke måtte betyde en forringet performance af arkiv og søge funktionaliteten i BZ5503.

Der var en enighed om at der ikke var nævneværdige forringelser af performance ved almindeligt brug af programmet og udvidelsen kunne derfor accepteres.

Der blev dog konstateret at der ved foldere indeholdende rigtig mange projekter var nogen ventetid før folderen blev vist.

Efter at have timet de enkelte dele af koden, har jeg konstateret at det er selve oprettelsen af IFilemeasurement og dermed udtrækningen af data fra det binære format der tager tid. Da dette er en helt essentiel del af udvidelsen kan dette ikke undværes, men der er dog givet et muligt løsningsforslag i Kapitel 9, Udvidelsesmuligheder.

9. Udvidelsesmuligheder

9.1. Fejlrapportering ved korruperede data

Som programmet er nu, rapporteres der ikke til brugeren hvis der findes data der ikke stemmer over ens med forventningerne. Dette er dog ikke et GridView specifikt problem og kan ikke løses blot ved at ændre i denne komponent.

For at kunne rapportere til brugeren, når der findes et projekt med fejlbehæftede data, er det nødvendigt først og fremmest at lave ændringer i IMeasurement. Disse ændringer skal laves så eventuelle fejl ved læsning af data rapporteres videre i systemet og de dermed kan rapporteres videre.

9.2. Performance forbedringer

Som det blev konkluderet i testen, så har skiftet fra ListView til GridView betydet et betydeligt performance tab. Dette tab skyldes ifølge målinger i koden den tid det tager .NET at foretage sikkerheds check på de filer der skal åbnes og tilgå disse.

Der er ikke umiddelbart nogen løsning på dette problem, men én mulighed ville være at starte med at load de oprindelige data ind i listen. Efter at disse data er loadet ind, hvilket ville tage den tid det tidligere har taget at åbne et arkiv, kan der startes en tråd der fylder nøgledata i listen.

Dette ville betyde, at man ville opnå den tidligere performance, men at man efter et kort øjeblik også ville have den nye udvidede funktionalitet til rådighed.

10. Konklusion

Der er implementeret en opgradering til BZ5503, som gør brugere i stand til hurtigt at identificere individuelle projekter i deres arkiver.

Løsningen benytter sig af såvel rå data som genkendelige overviews, til hurtigt at give brugeren et overblik. Desuden er alle metadata, der kan være vedhæftet et projekt eller en folder, trukket frem så de er lette at tilgå for brugeren.

Det er lykkedes at implementere funktionalitet, der tilfredsstillende alle krav stillet i kravspecifikationen, dog er der nogle problemer med performance i foldere med et stort antal projekter. Et forslag til løsning af dette er givet i kapitel 9, udvidelsesmuligheder.

Udviklingen af projektet har i alle faser haft brugeren i fokus og al funktionalitet konstant blevet evalueret ved hjælp af interne kunderepræsentanter. Dette har blandt andet betydet at der er tilføjet ekstra funktionalitet, som ikke i udgangspunktet var planlagt.

Løsningen i dens nuværende form er allerede frigivet til sælgere og kunder rundt omkring i verden. Endnu har ingen af disse meldt tilbage, men internt i huset er der stor begejstring for resultatet.

KAPITEL 11

11. Kildehenvisninger

[GEN]: Generics i .NET

<http://www.microsoft.com/danmark/msdn/dothenrik/article0205.aspx>

[SKA]: Rapport skabelon for eksamensprojekter på DTU.

<http://www2.imm.dtu.dk/teaching/thesis>

[DXG]: Developer Express – Extra Grid

<http://www.devexpress.com/Help/?document=XtraGrid>

[IEEE-830]: IEEE-830 standarden

Software Engineering, Principles and Practice

Hans van Vliet 2000

Forlag: Wiley

ISBN: 0-471-97508-7

KAPITEL 12

12. Bilag

12.1. Testrapport – Search

Testreport: Search - BZ 5503

Project Number: 60000128 - Logging Software for 2250

Archive: J:\2270\Arkiv\145_TestRappporter\Software
Item #: 2250 Software Package No. 11

Revision History:

Initials	Date	Revision	Remarks
PK75	2006-11-13	1	Created

Contents

PROJECT NUMBER: 60000128 - LOGGING SOFTWARE FOR 2250	1
ARCHIVE: J:\2270\ARKIV\145_TESTRAPPORTER\SOFTWARE	1
1 INTRODUCTION	2
2 RESULTS	3
2.1 Search.....	3
3 CONCLUSION	4
4 APPENDIX – TEST JOURNALS	4

1 Introduction

This document describes test of Search in BZ 5503.

Information about the test (fill in)

Tester	GWK
Date	060407
Duration of test	1 h
2250 serial no.	N/A
Version of BZ 5503	
Version of 2250 Software	
Windows version	
Servicepack	
Office version	
.NET Framework	
.NET Framework servicepack	
Other:	

Required accessories:

- CF-Card
- SD-Card
- CD-ROM with archive

In the following, please fill in the result of the test - either as OK, if the test passes, or as a reference to a Basic ENV Change Base report number, if the test fails.

Example of testresult:

Test Case	Result
Help is displayed when pressing the "?" icon	OK
Help text displayed for measurement parameters	Bug #8

2 Results

Copy "Test Archive SW Package 1 and 2" and "Test Archive SW Package 3" from the following folder to your PC in a folder of your choice (e.g. C:\BZ5503TestArchives):

J:\2270\Arkiv\145_TestRappporter\Software\Test Data\Archives

Copy "Test Archive Network" to N:\temp\BZ5503test (or similar)
Create a CD-ROM with "Test Archive CD-ROM", if not already existing.

Do not connect to the archives on j:\... - these are master archives for the test and should not be changed.

"Test Archive Archive SW package 8" on CD-ROM contains one user: 2250

2.1 Search

Test Case	Result
Insert the "Test Archive Archive SW package 8" CDROM into the CD-drive and connect the archive.	-
Select the root node right click and select search. Set type = "Project" and Node Type = "All"	-
Set Free Text = "Parameter LAS(SPL)" and Start Search. Check that Project 028 appears in results.	OK
Set Free Text = "BB=AC". Check that a number of results appear.	OK
Change Node Type to "Sound Level Meter". Check that results now only includes Sound Level Meter Projects.	OK
Change type to "User". Check that no results appear .	OK
Change Free Text to "2250". Check that all Users which contains 2250 in the name appears.	OK
Create a new user "2250TEST" in one of the archives on C-drive. Press "Start Search". Check that new user appears.	OK
Change type to "Annotation", Node Type to "All" and Free Text = "". Press Start Search. Check that alle annotations appears.	OK
Set Free Text = "Parameter LAS(SPL)" and Start Search. Check that Project 028 appears in results. Check that Setup Note from Project 028 appears by right clicking on result and press Locate.	OK
Right Click on Project 028's Setup Note and Press Edit. Check that the the Note is Read Only – not editable.	OK
Copy Project 028 to an archive on C-drive. Press Start Search. Check that the note appears.	OK
Locate the Setup Note related to the project on C drive. Check that this Setup Note is editable.	OK
Delete the Project 028 from the C-drive. Try and edit the Setup Note in Search View again. Check that an error message appears.	Bug #2100

Test Case	Result
Copy Project 028 to C-drive again. In Search set Type to "Project", Node Type to "All" and Free Text to "028". Press Start Search. Locate the newly copied Project 028.	
Right Click on Project 028 in search view. Select View. Check that the project appears in Data Viewer.	OK
Select Project 028 in Search View. Press Enter. Check that the project appears in Data Viewer.	OK
Right Click on Project 028 select Export. Export to 7820. Check that project appears in 7820.	

3 Conclusion

A minor bug report has been filed

4 Appendix – Test journals

Place all the measurement documentation (screen dumps, graphs, curves, printouts etc.) in this appendix for later reference. For documentation that are not possible to include in this appendix, include the file reference.

12.2. Testrapport – Archive Tasks

Testreport: Archive Tasks - BZ 5503

Project Number: 60000128 - Logging Software for 2250

Archive: J:\2270\Arkiv\145_TestRappporter\Software
Item #: 2250 Software Package No. 11

Revision History:

Initials	Date	Revision	Remarks
FLP75	2006-02-06	1	Created
FLP	2006-11-29	2	

Contents

1	INTRODUCTION.....	2
2	RESULTS	3
2.1	Managing Archives and Users.....	3
2.2	Organizing data in Archives	5
2.3	Viewing Data in Archive.....	8
2.4	Transfer data to Instrument	9
3	CONCLUSION	10
4	APPENDIX – TEST JOURNALS	10

1 Introduction

This document describes test of Archive Tasks (except Export) in BZ 5503.

Information about the test (fill in)

Tester	GWK
Date	050407
Duration of test	4½ h
2250 serial no.	Demo 34
Version of BZ 5503	
Version of 2250 Software	
Windows version	
Servicepack	
Office version	
.NET Framework	
.NET Framework servicepack	
Other:	

Required accessories:

- CF-Card
- SD-Card
- CD-ROM with archive

In the following, please fill in the result of the test - either as OK, if the test passes, or as a reference to a Basic ENV Change Base report number, if the test fails.

Example of testresult:

Test Case	Result
Help is displayed when pressing the "?" icon	OK
Help text displayed for measurement parameters	Bug #8

2 Results

Copy "Test Archive SW Package 1 and 2" and "Test Archive SW Package 3" from the following folder to your PC in a folder of your choice (e.g. C:\BZ5503TestArchives):

J:\2270\Arkiv\145_TestRappporter\Software\Test Data\Archives

Copy "Test Archive Network" to N:\temp\BZ5503test (or similar)
Create a CD-ROM with "Test Archive CD-ROM", if not already existing.

Do not connect to the archives on j:\... - these are master archives for the test and should not be changed.

"Test Archive SW Package 1 and 2" contains two users: 2250 and User 1

User: 2250: Colour Scheme is Indoor, All freq. weightings are AC

User: User 1: Colour Scheme is Arcade, All freq. weightings are AZ

"Test Archive SW Package 3" contains two users: 2250 and User 1

"Test Archive Network" contains one user: 2250

"Test Archive CD-ROM" contains one user: 2250

2.1 Managing Archives and Users

Test Case	Result
Connect to the archives above by creating a new archive with the same name in the same folder (e.g. Create a New Archive with the name "Test Archive SW Package 1 and 2" in the folder "C:\BZ5503TestArchives").	OK
Create a new archive "Test". This will be empty	OK
Right Click on the Test archive and select Create User. Select the name "Test User 1" and use default settings. The user has no data, but in the setting you have got the default templates.	OK
Select Data Type = Settings and edit the template CPB. Set Frequency Weighting Spectrum = A, close the editor. Press toolbar button Users. Select archive Test to create a new user in Test : Select the name Test User 2 and copy settings from an existing user, select Test User 1. Verify using Edit of CPB template, that Spectrum Weighting = A.	OK
Copy user 2250 from " Test Archive SW Package 1 and 2 " to " Test ". Verify all data and settings have been copied.	OK
Try to create a user with the same name as an existing user. Error message is given.	OK
Right click on 2250 in Test and select Delete. Verify the user is deleted.	OK
Select "Test User 2" in Test and press Delete. Verify the user is deleted.	
Right click on Test and select Disconnect Archive. The archive disappears from the list of archives.	OK

Test Case	Result
Connect to the archive Test by creating an archive with the name Test in the folder where the Test archive is located.	OK
Delete the archive Test . Verify Test is deleted from the disk.	OK
<i>Extend with your own test cases:</i>	

2.2 Organizing data in Archives

Test Case	Result
Connect a SD-Card and a CF-Card to the PC. All data in "Test Archive SW Package 1 and 2", "Test Archive SW Package 3", "Test Archive Network", "Test Archive CD-ROM" and the removable disks with SD-Card and CF-Card can be seen in Explorer.	OK
Navigate using Tab, Shift+Tab, Left, Right, Up, Down	OK
Create Job folders on a user in Test Archive SW Package 1 and 2, SD-Card and CF-Card using right click and create	OK
Create Job folders on a user in Test Archive Network, SD-Card and CF-Card using menu File-New-Job Folder	OK
Delete some Projects and Job folders on a user in an archive, SD-Card and CF-Card using right click and Delete	OK
Delete some Projects and Job folders on a user in an archive, SD-Card and CF-Card using Delete key	OK
Delete some Projects and Job folders on a user in an archive, SD-Card and CF-Card using Delete button in control panel	OK
Delete Annotations using delete key and Edit	OK
Add annotations using menu and context menu	OK
<i>Extend with your own test cases:</i>	

Move and Copy data –THIS IS ONLY MADE AS RANDOM TEST

Arc is an Archive, RD is a Removable Disk, CD is "Test Archive CD-ROM"

Select Measurement Data	From Arc1 to Arc2	From Arc1 to RD1	From RD1 to Arc1	From RD1 to RD2	From CD to Arc1	From CD to RD1
Move ^{*)} a single job or project using drag/drop	OK	OK	OK	OK	OK	OK
Move ^{*)} multiple jobs and projects using drag/drop	OK	OK	OK	OK	OK	OK
Move ^{*)} annotations using drag/drop	OK	OK	OK	OK	OK	OK
Right click cut ^{*)} /paste a single project	OK	OK	OK	OK	OK	OK
Right click cut ^{*)} /paste a single job folder	OK	OK	OK	OK	OK	OK
Right click cut ^{*)} /paste multiple jobs and projects	OK	OK	OK	OK	OK	OK

Select Measurement Data	From Arc1 to Arc2	From Arc1 to RD1	From RD1 to Arc1	From RD1 to RD2	From CD to Arc1	From CD to RD1
Right click cut ^{*)} /paste annotations	OK	OK	OK	OK	OK	OK
Copy a single job or project using CTRL drag/drop	OK	OK	OK	OK	OK	OK
Copy multiple jobs and projects using CTRL drag/drop	OK	OK	OK	OK	OK	OK
Copy annotations using CTRL drag/drop	OK	OK	OK	OK	OK	OK
Right click copy/paste a single project	OK	OK	OK	OK	OK	OK
Right click copy/paste a single job folder	OK	OK	OK	OK	OK	OK
Right click copy/paste multiple jobs and projects	OK	OK	OK	OK	OK	OK
Right click copy/paste annotations	OK	OK	OK	OK	OK	OK

To another user:

	From Arc1 to Arc2 on another user	From Arc1 to RD1 on another user	From RD1 to Arc1 on another user	From RD1 to RD2 on another user	From CD to Arc1 on another user	From CD to RD1 on another user
Move ^{*)} multiple jobs and projects using drag/drop	OK	OK	OK	OK	OK	OK
Copy multiple jobs and projects using CTRL drag/drop	OK	OK	OK	OK	OK	OK
Copy/paste annotations	OK	OK	OK	OK	OK	OK
Cut/paste annotations	OK	OK	OK	OK	OK	OK

*) Note: Moving data from an archive to a removable disk should always make a copy.

2.3 Organizing templates in an Archive

Test Case	Result
Select Data Type = Settings	OK
All settings in a user can be seen in Explorer (one Preferences and the Templates)	
Settings can not be displayed on the removable disks	OK
Navigate using Tab, Shift+Tab, Left, Right, Up, Down	OK
Delete a template using right click and Delete	OK
Delete a template using the Delete key	OK
Delete a template using the Delete button in control panel	N/A
The last template on a user cannot be deleted	OK
Check that Preferences and Template folder cannot be deleted	OK
Move a single template to another user using drag/drop	OK
Move a single template to another user using cut/paste	OK
Copy a single template to another user using drag/drop	OK
Copy a single template to another user using copy/paste	OK
Copy a single template from Test Archive CD-ROM to another user using copy/paste	OK
Copy Preferences to another user using copy/paste	
Check that settings cannot be copied to removable disks	OK
Open a Template on CDROM archive	#2121

Note: User 1 in "Test Archive SW Package 1 and 2" has Colour Scheme and Freq. weightings different from all other users.

2.4 Viewing Data in Archive

There have been made a separate for DataViewer.

Test Case	Result
Select Archive Task View.	N/A
View data from User: 2250 in Test Archive SW Package 1 and 2 . The data should match the description in the Ver 1.0 job folder.	N/A
View data from User: User 1 in Test Archive SW Package 1 and 2 . The data should match the description in the Ver 1.0 job folder.	N/A
View data from Test Archive Network	N/A
View data from Test Archive CD-ROM	N/A
View data from a removable disk	N/A
Edit a template from User: 2250 in Test Archive SW Package 1 and 2 .	N/A
Edit a template from Test Archive Network .	N/A
Cannot Edit a template from Test Archive CD-ROM .	N/A
Extend with your own test cases:	

2.5 Transfer data to Instrument

Test Case	Result
Select Archive Task To Instrument. Connect 2250 to the PC. Make sure you have a user in the archive that doesn't exist on the 2250.	OK
Select Measurement Data and a user in the archive The Archive or User can be transferred.	OK
Select a job folder and/or project of a user in Test Archive SW Package 3 (the user also existing in 2250). Transfer the data. Verify that all data has been transferred to the instrument. Data with the same name in the archive as in the instrument will get a new name.	OK
Select a job folder and/or project of a user in Test Archive CD-ROM (the user also existing in 2250). Transfer the data. Verify that all data has been transferred to the instrument. Data with the same name in the archive as in the instrument will get a new name.	OK
Select a job folder and/or project of a user in Test Archive Network (the user not existing in 2250). Transfer the data. Verify that a user has been created on the instrument and all data has been transferred to the instrument. The settings of the user has also been transferred.	OK
Select a job folder and/or project of a user in Test Archive CD-ROM (the user not existing in 2250). Transfer the data. Verify that a user has been created on the instrument and all data has been transferred to the instrument. The settings of the user has also been transferred.	OK
Select Settings in Test Archive SW Package 1 and 2 User 1 Transfer Preferences Verify that preferences only has OK been transferred	OK
Select Settings in Test Archive CD-ROM Transfer Preferences Verify that preferences only has been transferred	OK
Select Settings Test Archive SW Package 1 and 2 User 1 Transfer a single template Verify that this template only has been transferred	OK
Select Settings Test Archive CD-ROM Transfer a single template Verify that this template only has been transferred	OK
Select Settings Test Archive Network Transfer Templates folder Verify that all templates has been transferred	OK
Extend with your own test cases:	

3 Conclusion

Only found one error when trying to open a template on CD ROM.

4 Appendix – Test journals

Place all the measurement documentation (screen dumps, graphs, curves, printouts etc.) in this appendix for later reference. For documentation that are not possible to include in this appendix, include the file reference.

