



Allstate®

You're in good hands.

Enterprise Business Intelligence Xelsius 2008 and Beyond

Jeffrey Strauss – BI Architect
Nick Knol – Software Engineer

3rd quarter 2008



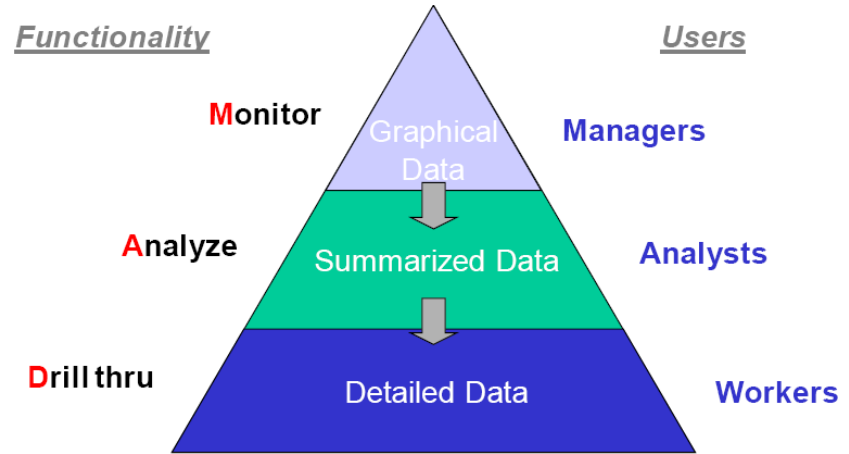
Xcelsius value proposition

Solution for Building and deploying Visually Interactive Apps

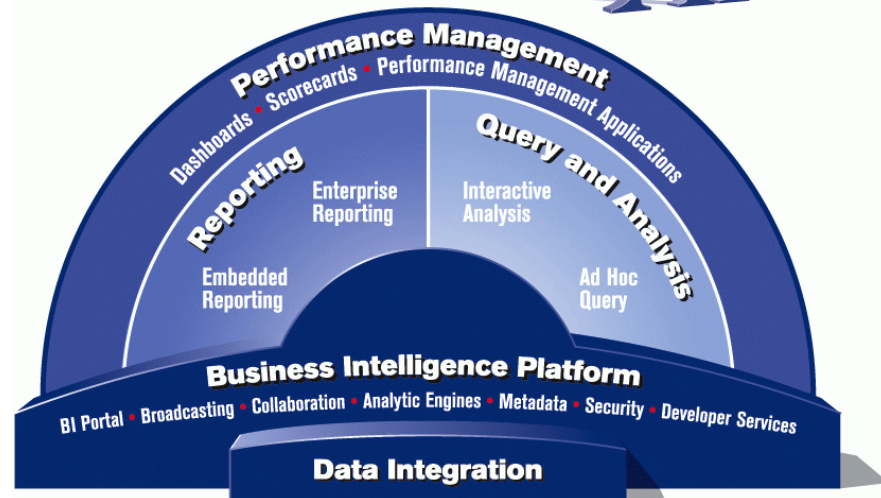
- ◆ Increase user adoption by ease of use, interactivity, and sophistication
- ◆ Maximize ROI by best use of technology



Timeliness, inquiry and analysis



BusinessObjects™ XI



Topics

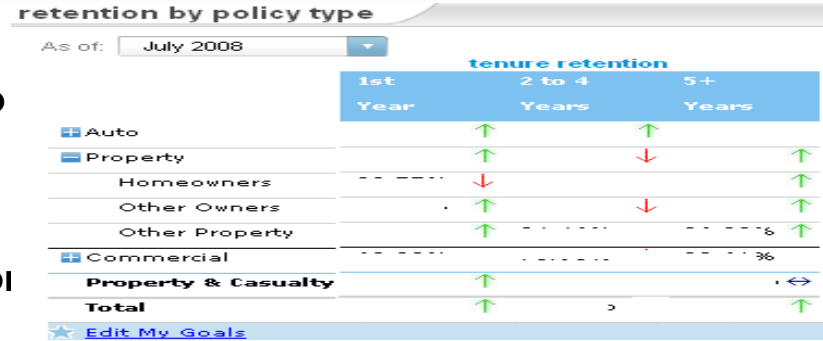
- ◆ **Xcelsius primer**
 - What is it and where does it fit
 - Architecture introduction
- ◆ **How has Allstate deployed Xcelsius**
- ◆ **Xcelsius in action**
- ◆ **Integration points / Architecture in-depth**
- ◆ **Flex interface highlights**
- ◆ **Best Practices / Tips**
- ◆ **Q&A**

What is Xcelsius?

Solution for Building and deploying Visually Interactive Apps

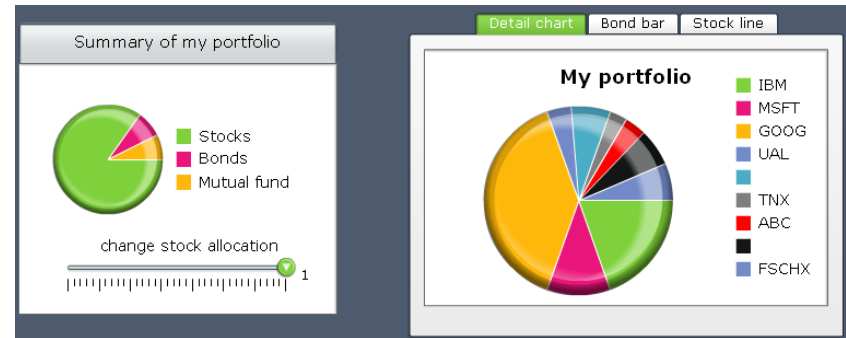
Visual eye catching

- ◆ Turn ordinary data into charts, graphs, tables
- ◆ Enable alerts for performance orientation



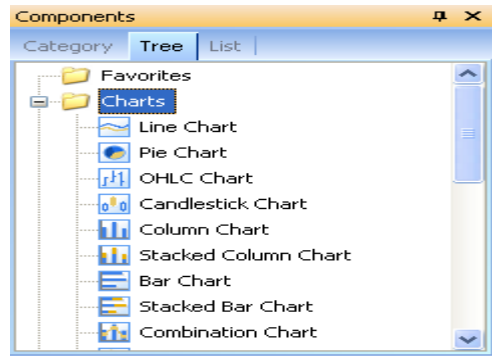
Interactive / Sophisticated

- ◆ Consider different outcomes through “what-if” analysis
- ◆ Enable different views without server round-trips



Agile “drag-drop” results

- ◆ Shorten the time required to reach final decisions
- ◆ Over 70 standard widgets, 9 themes, and 30 color schemes

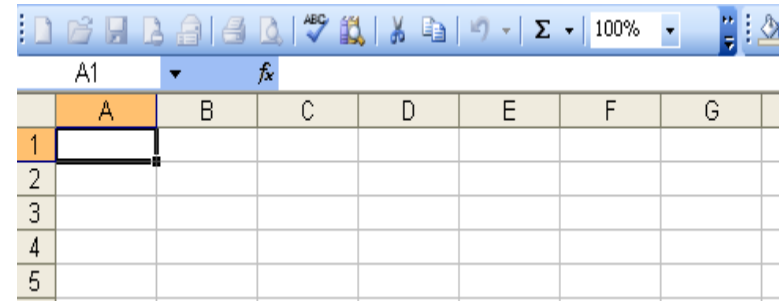


What is Xcelsius?

Solution for Building and deploying Visually Interactive Apps

Analytic summary tool

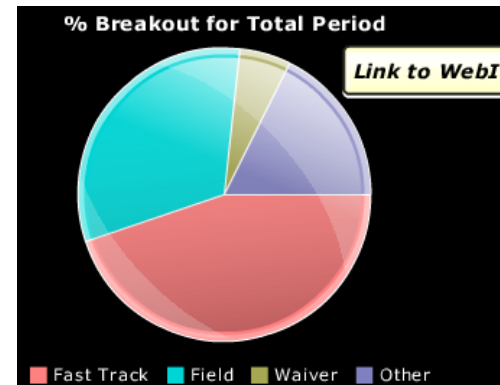
- ◆ Leverage the power of Excel formulas
- ◆ Summary – YES but Detail – No. The tool does not have a built in aggregator



| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | | | | | | | |
| 2 | | | | | | | |
| 3 | | | | | | | |
| 4 | | | | | | | |
| 5 | | | | | | | |

Access gateway to details

- ◆ Drill down to your Crystal or webi documents
- ◆ Leverage Live Office services to access Crystal historicals



Integrated into BOBJ suite

- ◆ Leverage the BOE framework for integration with universes, security, and links into webi and Crystal

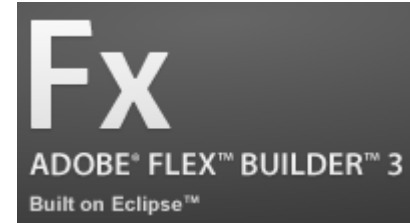


What is Xcelsius?

Solution for Building and deploying Visually Interactive Apps

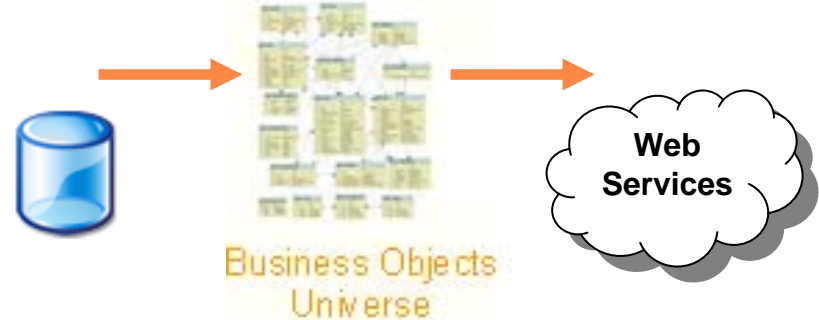
Open – Extensible development

- ◆ Develop custom widgets



Open – Many data sources

- ◆ Leverage your universe through web services
- ◆ Leverage Excel data
- ◆ Flash variables



Open – Many output types

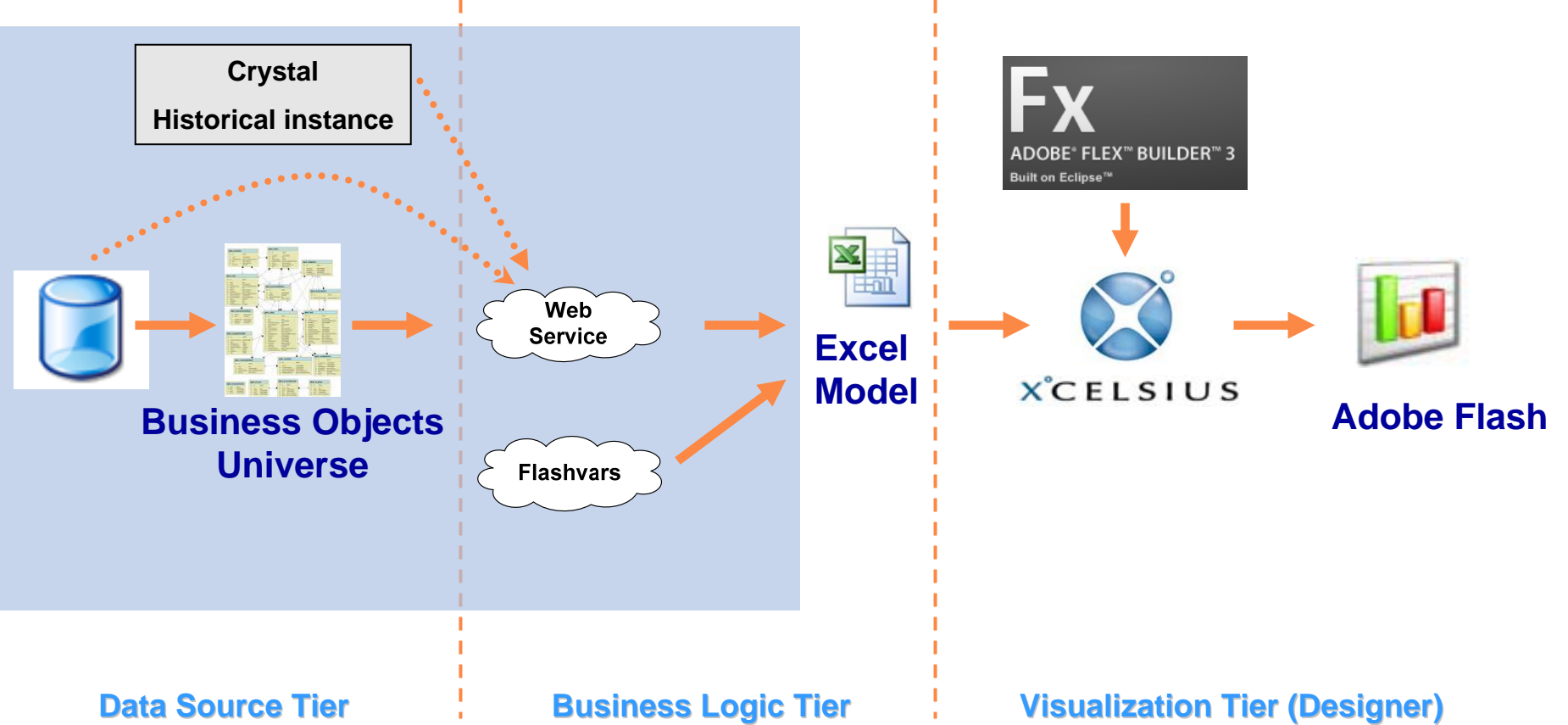
- ◆ Infoview
- ◆ Office
- ◆ Adobe PDF
- ◆ Crystal reports 2008
- ◆ Portals
- ◆ App integration



Adobe Flash

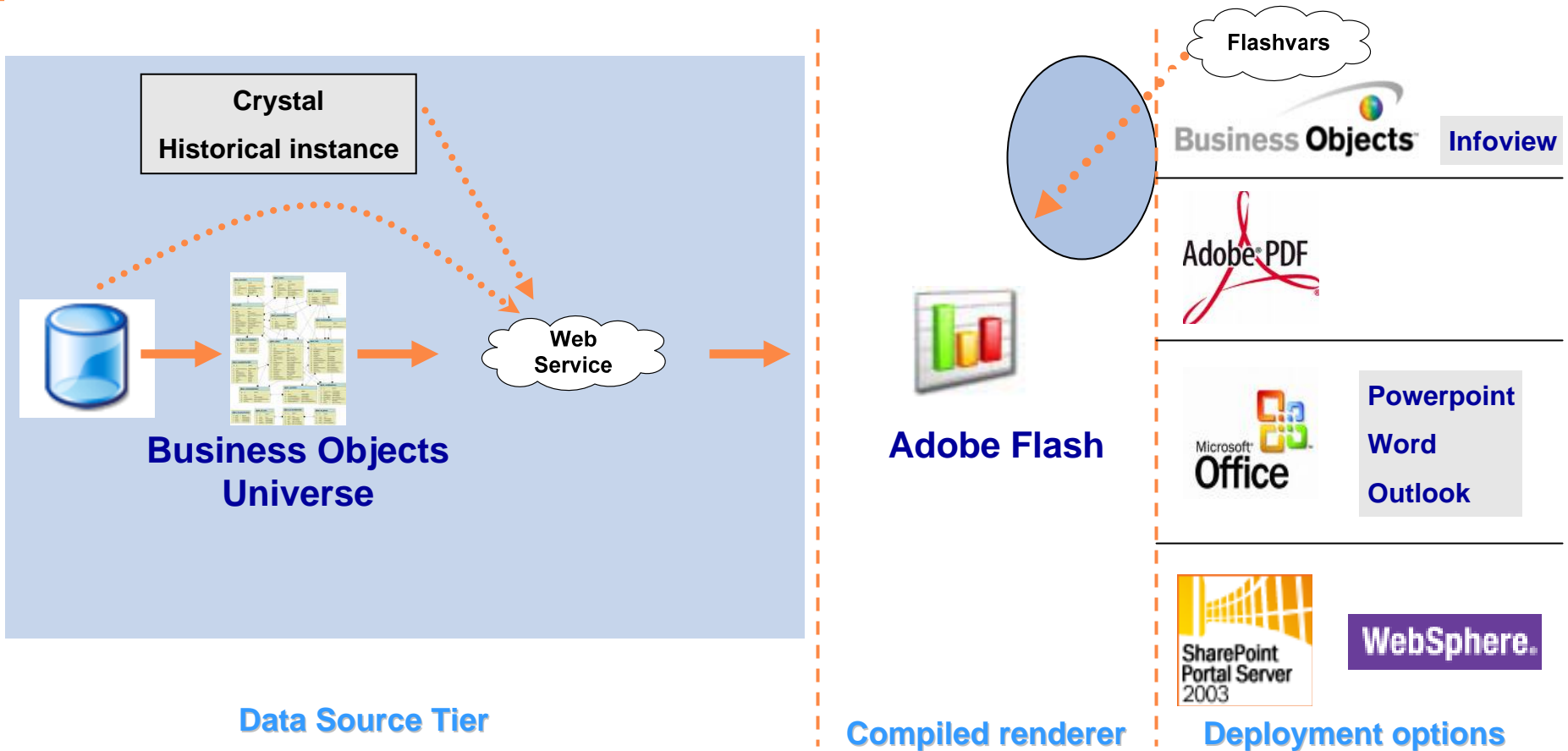
Xcelsius architecture - Introduction

Architecture - Design time



Xcelsius architecture - Introduction

Architecture – Deployment time



Xcelsius @ Allstate – Agency Distribution

- ◆ Our largest dashboard application (MAR) is deployed to our agency force ~ 14,000
- ◆ Integrates with Websphere portal and TAM
- ◆ Runs over the extranet
- ◆ Uses custom controls
- ◆ Drills to detail reports

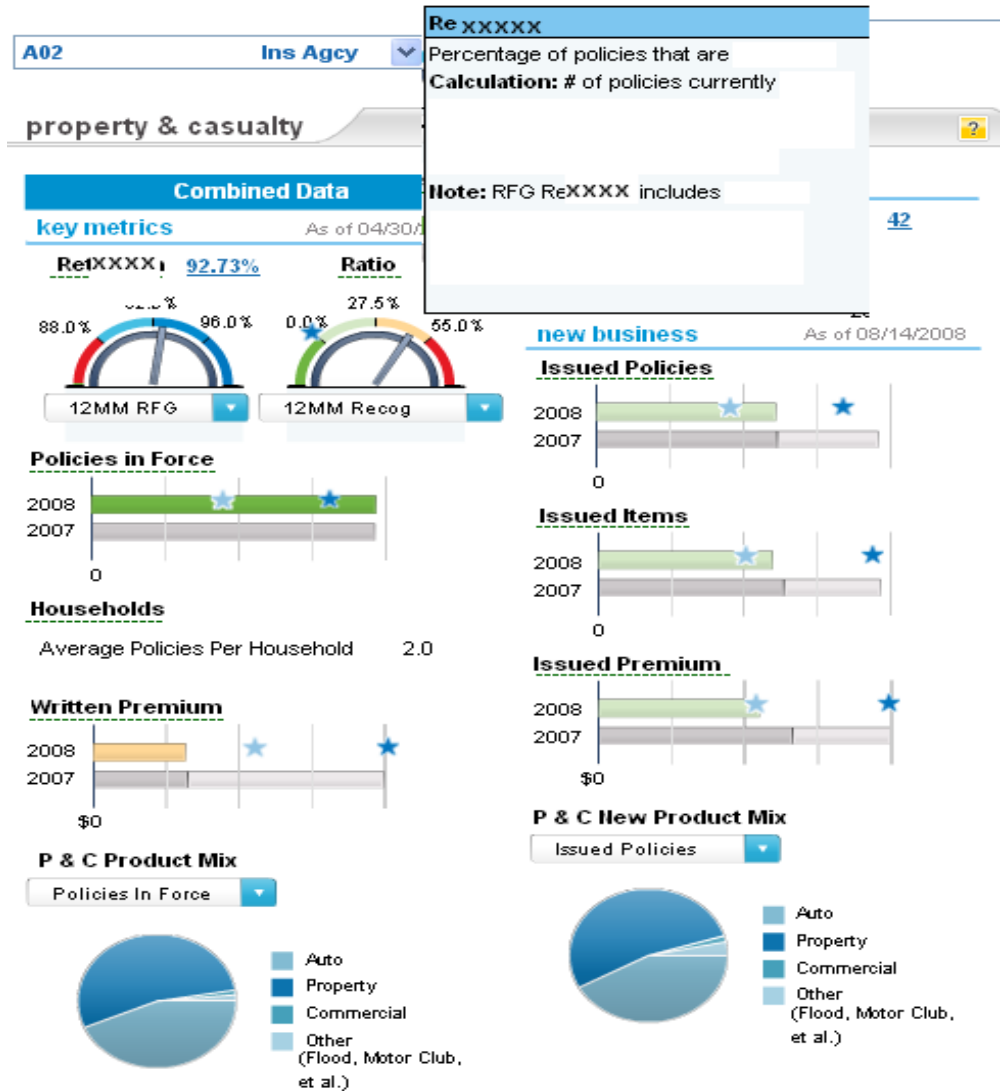
retention by policy type

As of: July 2008

tenure retention

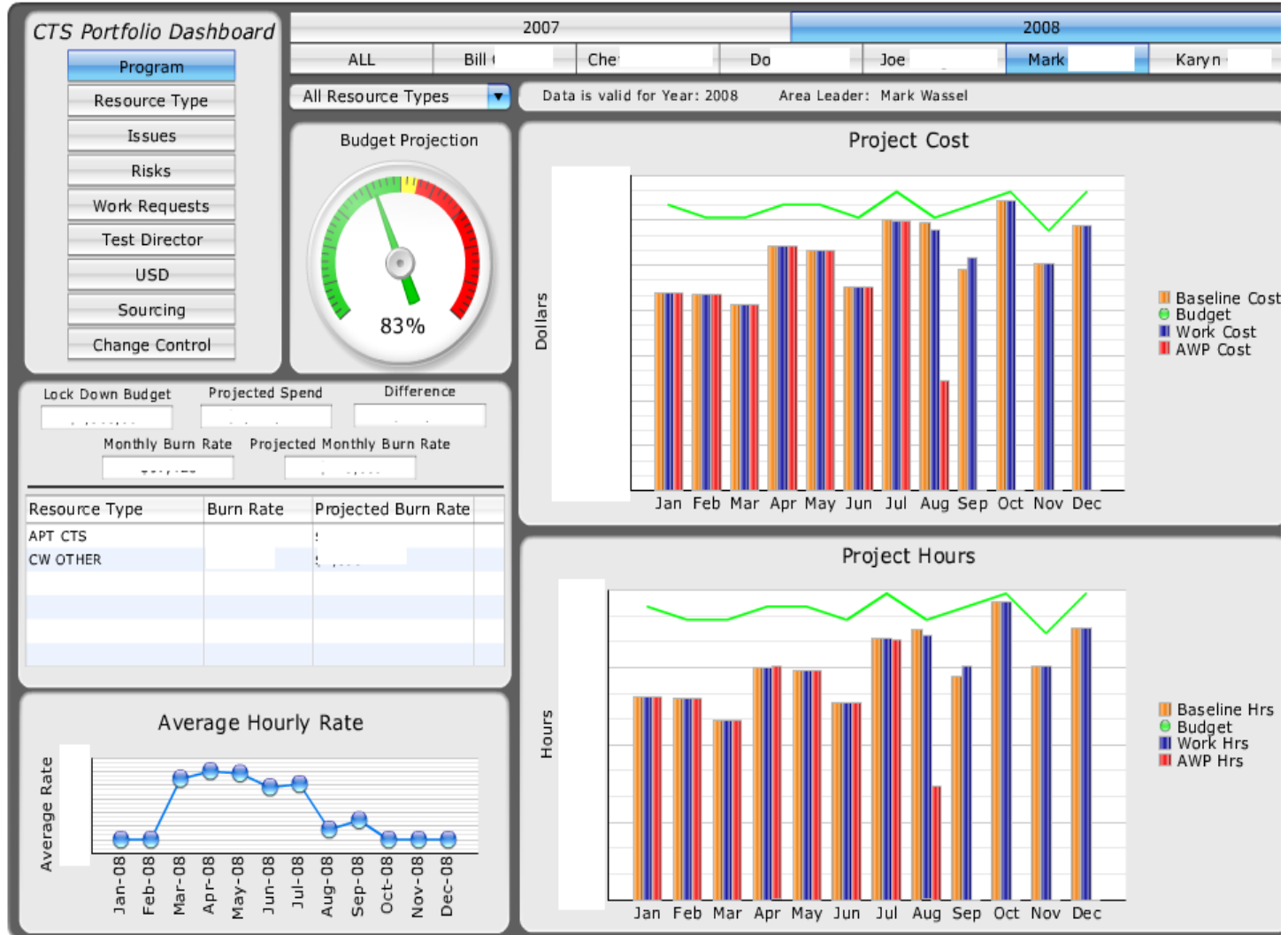
| | 1st Year | 2 to 4 Years | 5+ Years |
|--------------------------------|----------|--------------|----------|
| Auto | ↑ | | ↑ |
| Property | ↑ | | ↓ |
| Homeowners | ↓ | | ↑ |
| Other Owners | ↑ | | ↓ |
| Other Property | ↑ | | ↑ |
| Commercial | | | % |
| Property & Casualty | ↑ | | ↔ |
| Total | ↑ | | ↑ |

★ [Edit My Goals](#)



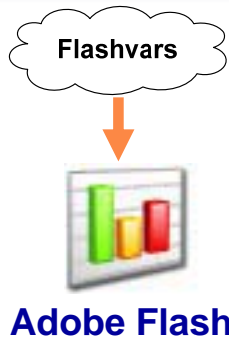
Xcelsius @ Allstate – Claims Technology Services

- ◆ Operations program mgmt application
- ◆ Accessed by Infoview
- ◆ Security controlled by Active Directory
- ◆ Uses Query as a web service

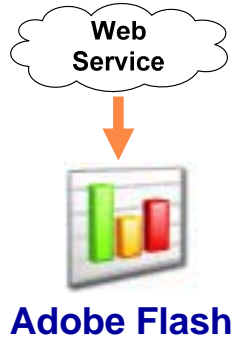
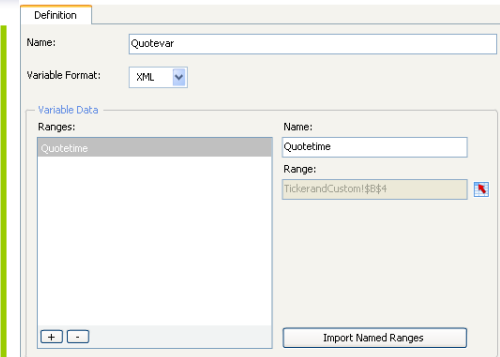


Live Xcelsius Demo

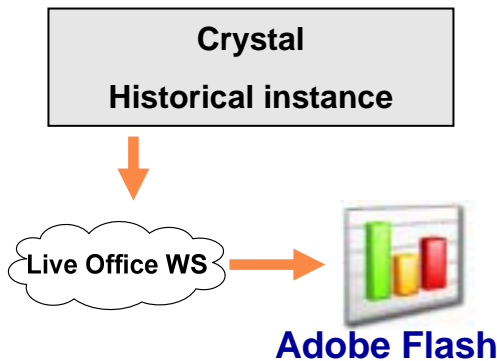
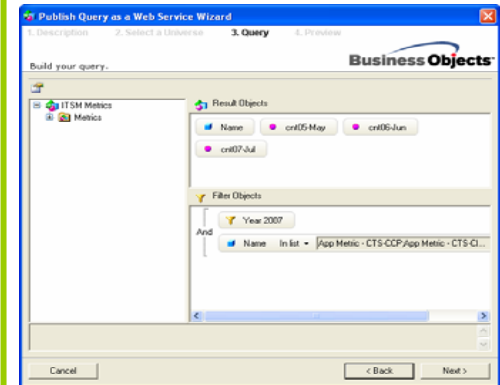
Integration Points – Primary In



- ◆ Pass singleton variables in from outside applications to the Flash onLoad
- ◆ Good for row level security



- ◆ Pass query results during execution
- ◆ SOAP over HTTP
- ◆ Use QaaWs or custom
- ◆ Requires single schema & no soap header



- ◆ Pass scheduled report instances during execution
- ◆ Good for consumption of existing report data



Integration Points – Primary Out

Crystal Report part data consumer

- ◆ Integrate dashboard components into Crystal2008

Definition

Row Headers:

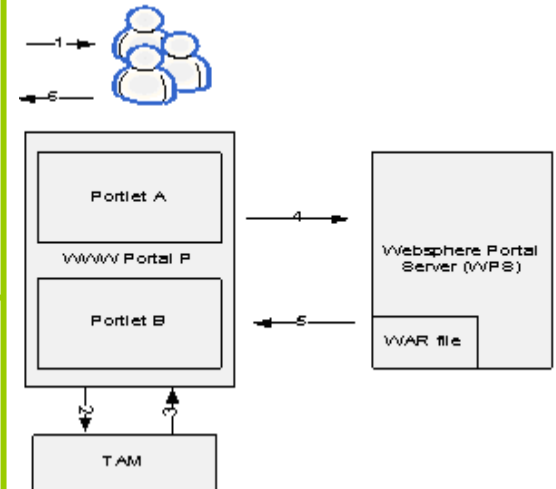
Column Headers:

Data:

Description:

Portal consumers / providers

- ◆ **Consumer** – Pass singleton / row / table from portal technology to Flash during execution
- ◆ **Provider** – Pass singleton / row / table from Flash to portal technology during execution
- ◆ Often triggered by an event



Websphere WAR file

- ◆ **War file** – Communication metadata file that manages wiring relationships between portlets

Integration Points – Less known

FS Command

- ◆ **Flash command** – Interfaces with the HTML browser container
- ◆ **Primary Use case** – Trigger a javascript event based on an action on the dashboard

Definition Usage

Name: Connection 1

Command:

Parameters:

LCDS

- ◆ **Live Cycle Data Services** – Adobe data monitoring interface
- ◆ **Primary Use case** – Real time refresh of dashboard based on app update to the data

Definition

Available Data Destinations

| Id | Description |
|----|-------------|
|----|-------------|

Advanced >>

Destination Data

| Data Element | Data Type | Shape |
|--------------|-----------|-------|
|--------------|-----------|-------|

Cross domain.xml

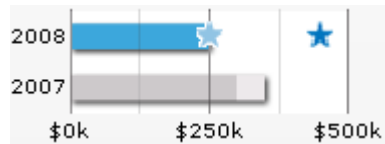
- ◆ Security file that needs to be at the root of web server
- ◆ Required for data service calls from Flash to data server
- ◆ Pointer to data services tier

Integration Points – Design



- ◆ **Develop custom widgets (2 primary components):**
 - Control
 - Property Sheet
- ◆ **Integrate using the Flex SDK**

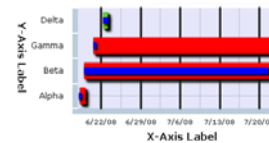
Progress bar



Expand / Collapse

+ <Parent1... <Field>
+ <Parent2> Field & < > " "
<Parent3> ↔ <Field>

Gantt



Four stars



- ◆ **What is Flex?**
- ◆ **Why Should You Care**
- ◆ **Problems and Fixes**
- ◆ **Summary**

- ◆ **Open source technology from Adobe in 2004**

- ◆ **Programming model for ActionScript (think HTML and JSP)**

- ◆ **IDE: Flex Builder 3**
 - Standard: \$249
 - Professional: \$699
 - Free with your child's school ID!

The Flex Part – Why Should You Care

- ◆ **Custom Flex Components for XCelsius**
- ◆ **Can add needed functionality to an XCelsius model**
- ◆ **Simplify complex models into one component with easy Excel interface**
- ◆ **Addresses gaps in Look and Feel**

- ◆ **Back to writing code, D'oh!**
 - But really should only be one coder, forces modularity

- ◆ **Tricky API, learning curve**
 - Good manuals, examples provided by BO

- ◆ **Can be a debugging nightmare**
 - Flashtracer (Firefox add-on)

◆ Custom Flex Components Summary

- If you need more features
- If you need a certain look
- Your programmers are getting bored/whiny

Live Flex Demo

Best practices

◆ Organized Excel model

- Think about long term ease of understanding and maintenance
- Segment by type (external variables, external data, control types)

| | A | B | C | D | E | F |
|----|---------------|--------|---------------|-----------|------|---|
| 1 | External data | | | | | |
| 2 | In | | | | | |
| 3 | Flashvar | | | | | |
| 4 | Name | Usage | Default value | Input to: | | |
| 5 | Quotedelay | Ticker | 10 | QuoteVar | | |
| 6 | Web Service 1 | | | | | |
| 7 | Name | | | | WSDL | |
| 8 | | | | | | |
| 9 | | | | | | |
| 10 | | | | | | |
| 11 | | | | | | |

◆ Query results

- Use Xcelsius as a summary visualization tool...Otherwise there will be too many detail bindings
- Perform aggregations and calcs within queries. Link to the details using a Crystal report
- Never bind a visualization directly to the result. Rather abstract the binding by a workbook link
- Do not map individual XML fields directly to the model. Rather map the parent folder to a range
- Do not overfetch, but do not underfetch. Find the right balance to decrease server roundtrips

◆ Visualizations

- Do not hardcode labels. Rather reference the model which will optimally read from Flashvars or WS
- Plan the visualization with consistent fonts, control types, color schemes
- Name each control within the object browser and group common controls by functional area
- Include a toggle and dynamic visibility for help text

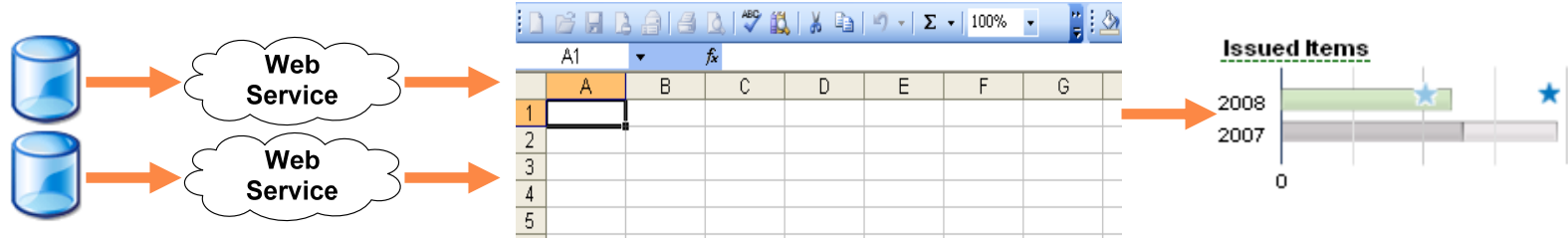
◆ Distribution

- Publish the compiled swf to a central location so that updates can be kept in sync
- Publish the source xlf to a controlled code repository for safe keeping and future maintenance

Tips

◆ Data sourcing for controls

- Think of the Excel model as the canvas data provider
- In effect, the control can be sourced from disparate services (i.e. plan, actual, goal)



◆ Minimize the output packet size

- Plan out the data services model and visualizations
- Try not to rename tabs after visualizations are linked to the model
- Always grab controls from the component browser. Do not copy controls on the canvas

◆ Custom controls within Flex

- When compiling on Flex 3, use the Flex 2 compiler SDK
- Bind controls using Actionscript. MXML works inconsistently
- Publish the source to a controlled code repository for safe keeping and maintenance

◆ Query as a Web Service and row level security

- This connects using Enterprise authentication
- Find a way to securely pass the userID to the universe which will use standard row level security

Conclusion and key take aways

- ◆ Xcelsius has enabled Allstate to standardize on one platform for dashboard creation
- ◆ Remember long term. Plan your inputs and outputs for your dashboard wisely in order to ease future maintenance
- ◆ Remember short term. Plan user adoption intuitiveness, ease of use and required calls back to the server
- ◆ Paint the canvas with remembering short-term, and long-term, but also use your imagination



Questions

◆ Questions

- Jeff Strauss, BI Architect, Allstate
- Nick Knol, Software Engineer, Allstate

◆ Contact information

- Jeffrey.Strauss@Allstate.COM
- 847-402-3027

- Nick.Knol@ALLSTATE.COM
- 847-402-7203

