

JavaBeans Activation Framework

Gary Murphy

Hilbert Computing, Inc.

**Colorado
Software Summit**



Hilbert Computing, Inc.

email: glm@hilbertinc.com

home page: www.hilbertinc.com

phone: 913-780-5051

The Goal

- **Create an awareness of this framework**
 - JAF is a standard Sun extension
 - Current release is version 1.0.1
 - Usually associated with JavaMail, but it is more general-purpose.

Sun Says...

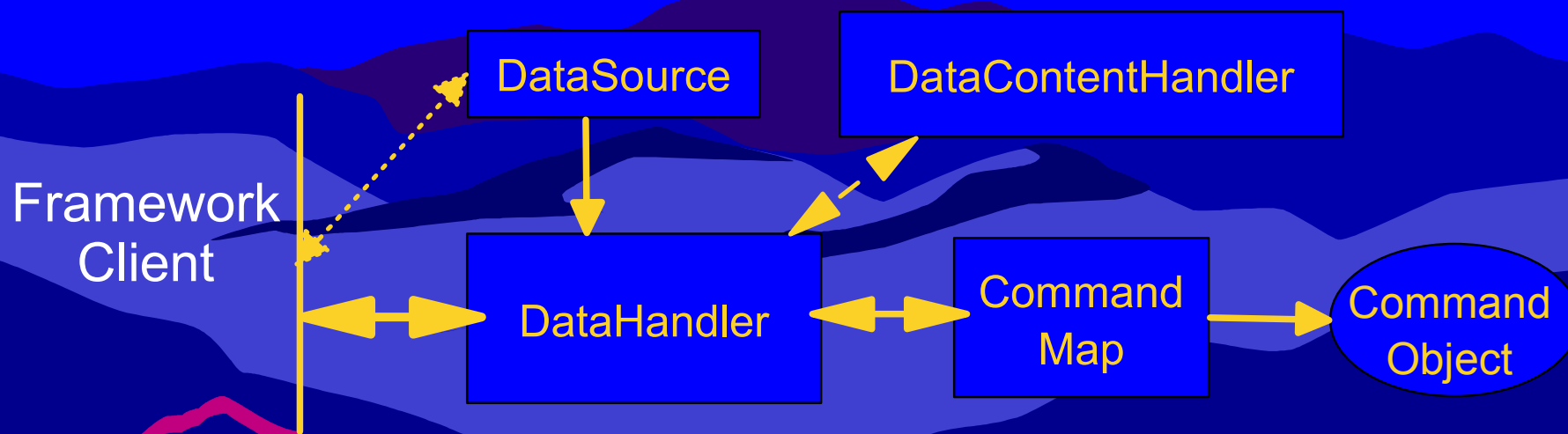
- **The JavaBeans Activation Framework implements the following services:**
 - It determines the type of arbitrary data
 - It encapsulates access to the data
 - It discovers operations on a particular type of data.
 - It instantiates the software component that corresponds to the desired operation on a particular piece of data

I Think of It As...

- ... a generalized data-driven factory.
 - Based on the MIME-type, it can determine what the data is.
 - Based on your configuration files, it can determine what “commands” are available to manipulate the data. For example:
 - Run — Load the data and execute it
 - View — Display the data using Swing components
 - Decrypt — A command I created to decrypt the data from a user-defined MIME-type
 - Based on the MIME-type and the command, it will construct a JavaBean to act on the data.

Graphical Overview

- **Five Primary Components:**
 - DataSource
 - DataContentHandler
 - DataHandler
 - CommandMap
 - CommandObject



DataSource Interface

- **Used to access the underlying data**
 - **Implements access to input and output streams, content type, and the name of the data.**
 - **Sun provides with JAF:**
 - **FileDataSource** — Access file system objects
 - **URLDataSource** — Access data from a URL
 - **Sun provides with Mail:**
 - **MimePartDataSource** — Access MIME data within an e-mail.

Note: Do not open the output stream in a data source unless you intend to write to it. In some cases, for example the FileDataSource, this will overwrite the contents of the data.

DataHandler Class

- Encapsulates type-to-command object binding
- Provides methods to act on the data. Preferably constructed with a DataSource object.

DataContentHandler Interface

- **Used to convert InputStreams into objects**
 - Used to map into the Transferable interface.
 - Uses DataFlavors to represent the data types that it can access

CommandMap Interface

- Uses the DataHandler MIME type to list the available commands.
- Maps MIME types and commands to CommandInfo objects.
 - Sun provides with JAF:
 - MailcapCommandMap — Implements a commandmap based on ‘mailcap’ files (RFC 1524)

CommandInfo Class

- Represents commands in an underlying registry
- Beans can be constructed from the information in the CommandInfo

CommandObject Interface

- Items must be of this to be “activated” by JAF
- Simple interface with one method:
 - `setCommandContext(...)`

DataContentHandler-Factory

- Can be used to map MIME types to DataContentHandlers
- Simple interface with one method:
 - createDataContentHandler(...)

Demo and Wrap-Up

- There are a lot of parts
- Let's look at some sample code...

- Questions?