



MAF3

The story so far
&
Architectural concepts

Paolo Quadrani

BioComputing Competence Centre (B3C)

p.quadrani@scsolutions.it

What is MAF?



<http://www.openmaf.org>

The **Multimod Application Framework** (MAF) is a freely available open source framework for the rapid development of computer aided medicine applications. It provides **high level components** that can be easily **combined** to develop a vertical application in different areas of scientific visualization.

Developed by a group of institutions including the BioComputing Competence Centre, The Rizzoli Institute, the CINECA Supercomputing centre, and the University of Bedfordshire

2000: MAF1

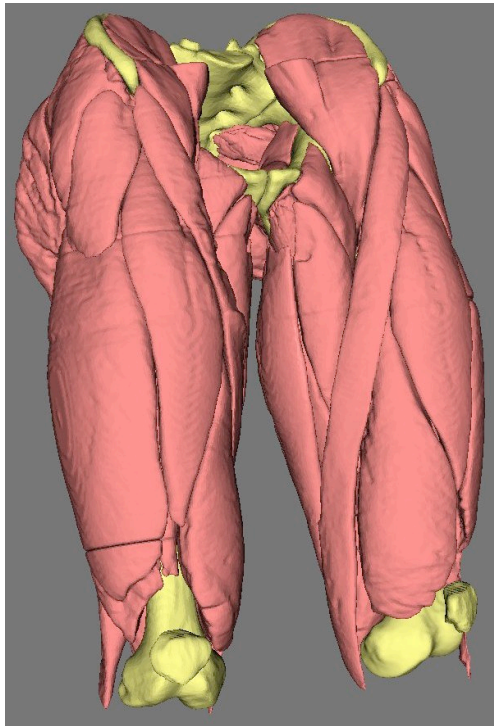
- Multimod Application Framework
 - Software application framework
 - Rapid development of computer aided medicine
 - applications requiring fusion of heterogeneous data
 - Applications capable of:
 - import any biomedical dataset
 - Register in space and time data from different imaging and instrumentation modalities
 - Provide a unified software environment for data processing, data fusion, and interactive visualisation

2004: MAF2

- CINECA develops the core of MAF version 2 (MAF2) with BSD licence
- The OpenMAF open source project is launched
- CINECA re-targets MAF as a general purpose framework
- Extreme Programming software development
- QA Infrastructure: Automatic source code building, dashboards, unit testing, memory leaks monitoring, coding convention checks...

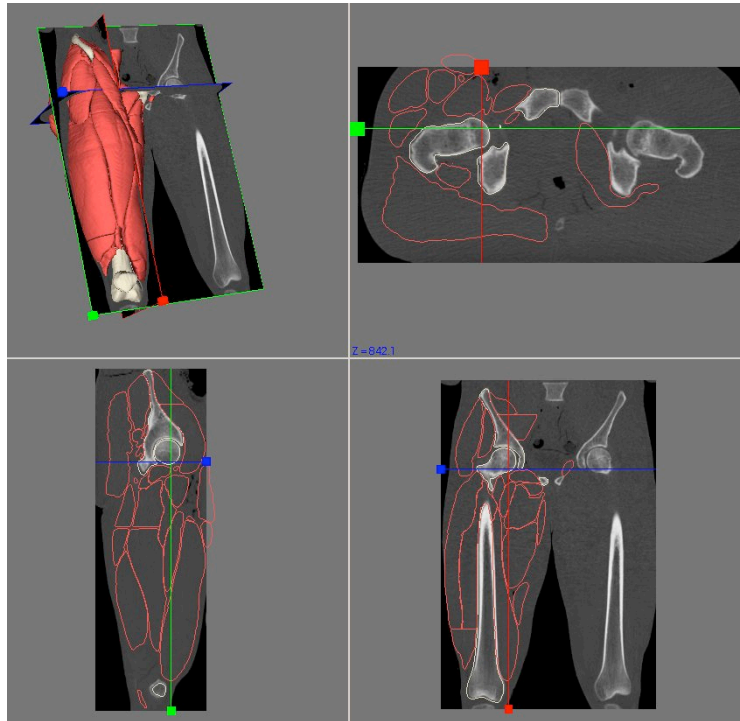
The Metaphor

Representation



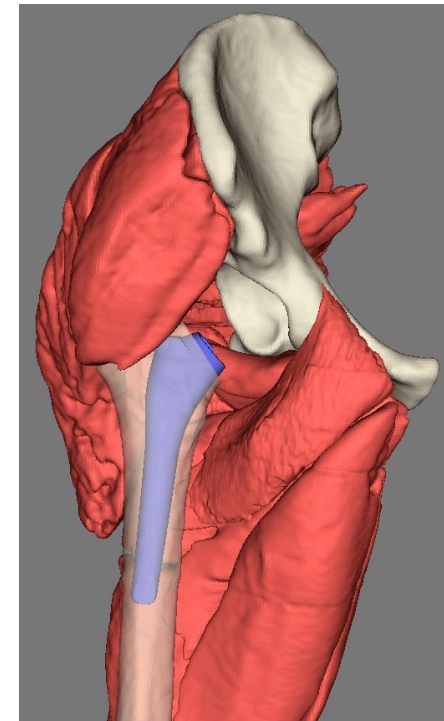
VMEs

Examination



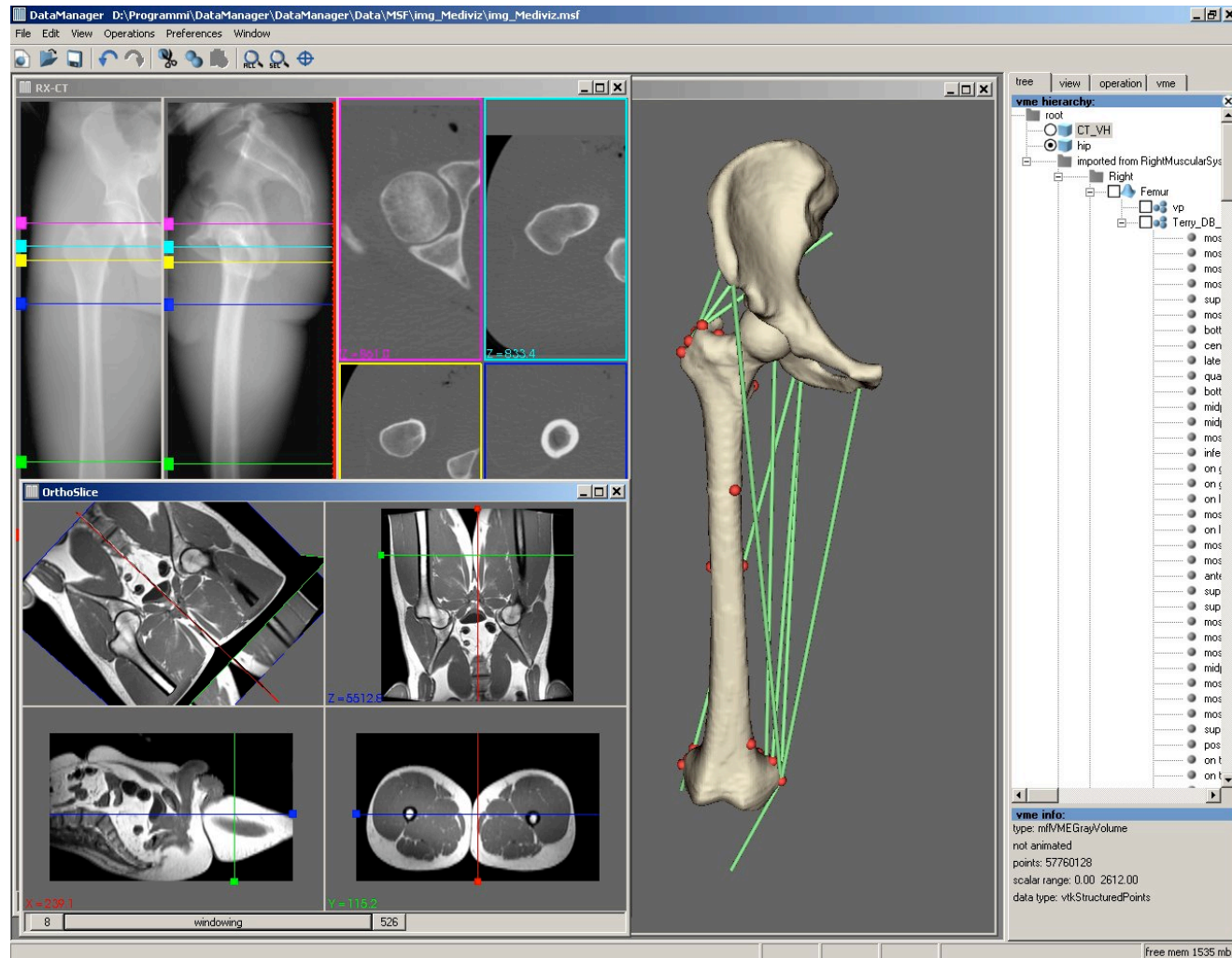
Views

Alteration



Operations

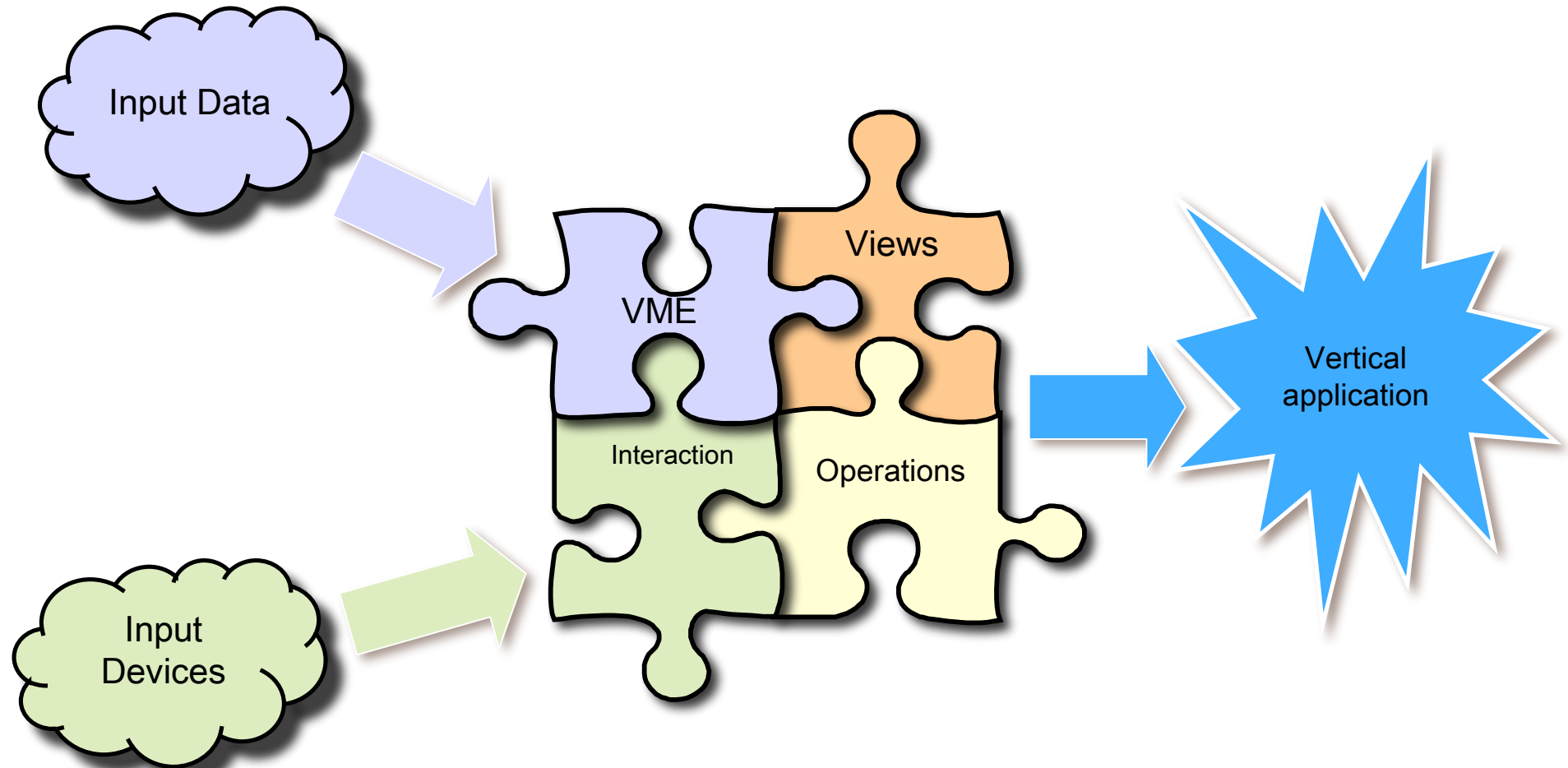
Data Fusion





FUNCTIONAL DESCRIPTION OF MAF2

MAF2 Architecture



From MAF2 to MAF3

- MAF2 main issues:
 - Monolithic architecture
 - Missing of plug-in extension mechanism
 - Missing of scripting integration
 - Missing support for Multithreading
 - ...
- MAF3 will fill these gaps (hopefully 😊)

MAF3 Main features



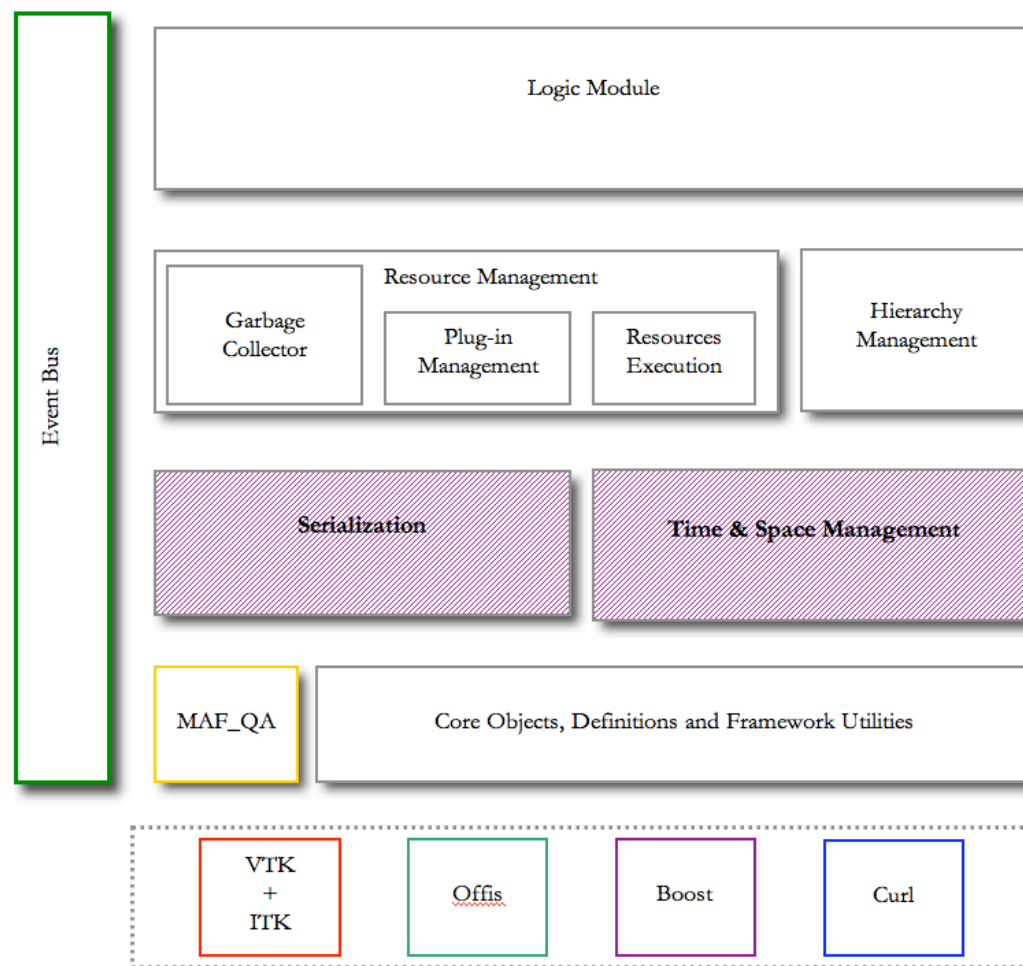
- Modular architecture
- Extensible through plug-ins
- Modules will have a Facade
- Modules will be scriptable having well defined public API
- Multithreading Support

Modular Architecture



- Foundation libraries
- Core Module
- Event Communication Bus
- Serialisation module
- Time & Space Management Module
- Data Resource
- Resource Management Module
- Hierarchy Management Module
- Logic Module

MAF3 Block Diagram



Foundation libraries

- MAF3 will rely on top of the following precompiled binary libraries:
 - VTK
 - ITK
 - Boost
 - Curl
 - [Widget set Library: Qt 4.5, wxWidget]

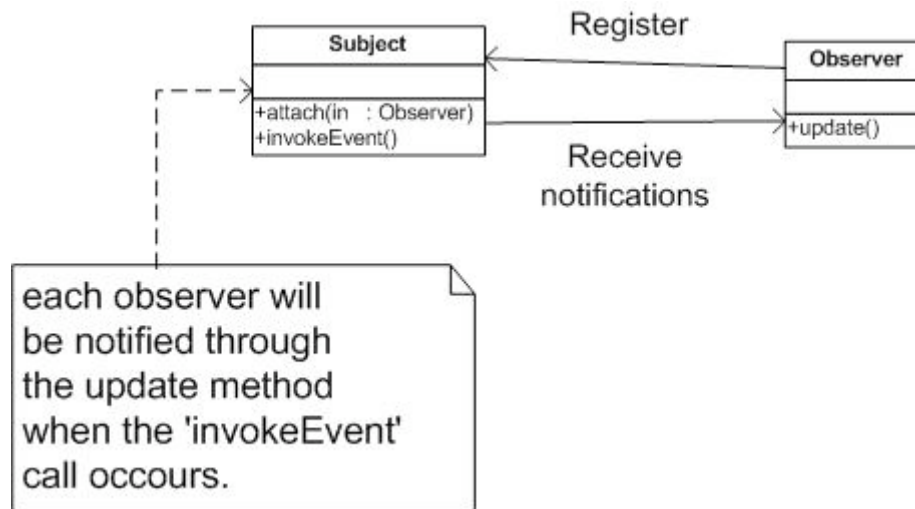
Core Module



- Defines core objects like
 - ▣ mafObject
 - ▣ the ID provider which generates unique IDs needed by MAF Objects and by events.
- Defines UNICODE switches for basic data types like mafChar
- Defines framework utilities for example
 - ▣ Thread creation & management

Event Communication Bus (1/2)

- Define the communication mechanism used by the framework
- Based on Observer pattern



Event Communication Bus (2/2)



- Allows objects registering as event observer.
- Notify all the observer when an event comes
- Has logging functionalities.

Serialisation Module



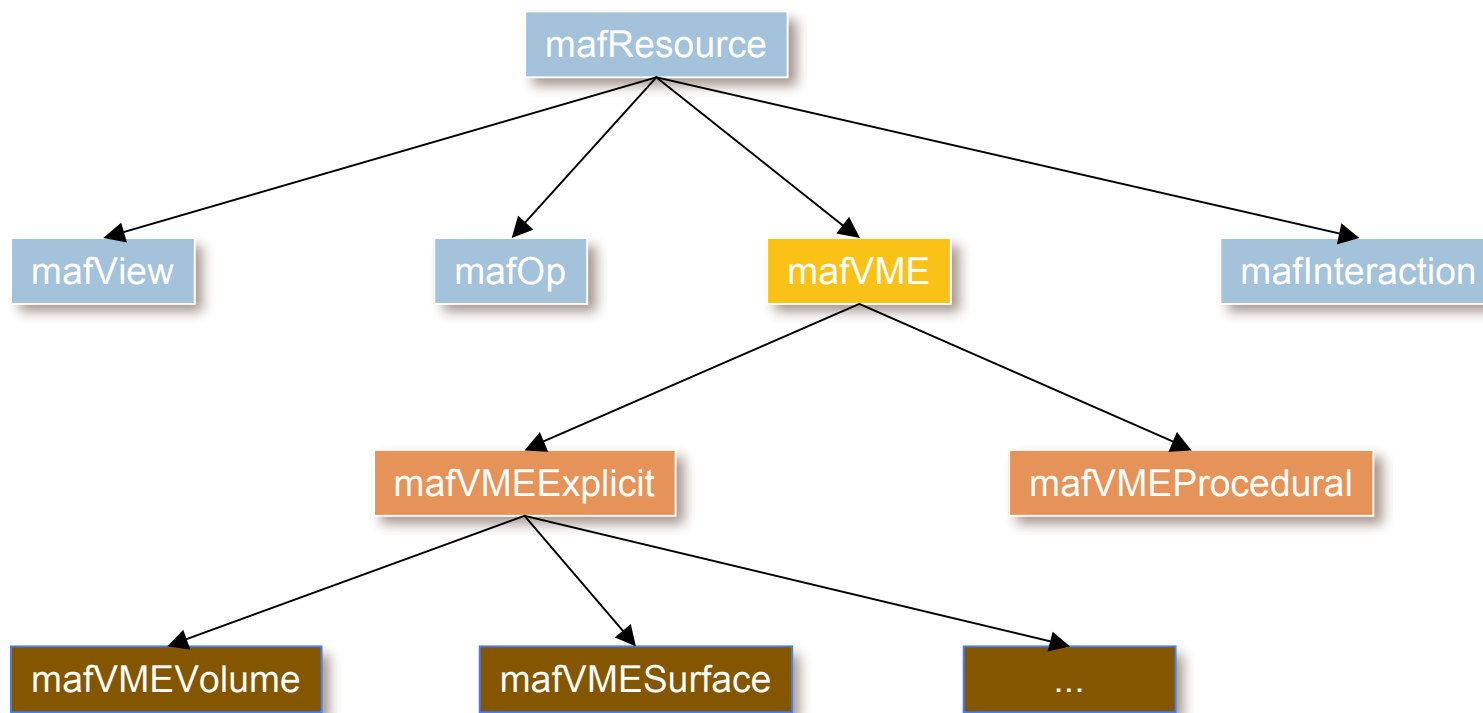
- Allows writing objects into a storage system which can be
 - ▣ File system
 - ▣ Database
 - ▣ Web remote storage

Time & Space Management Module

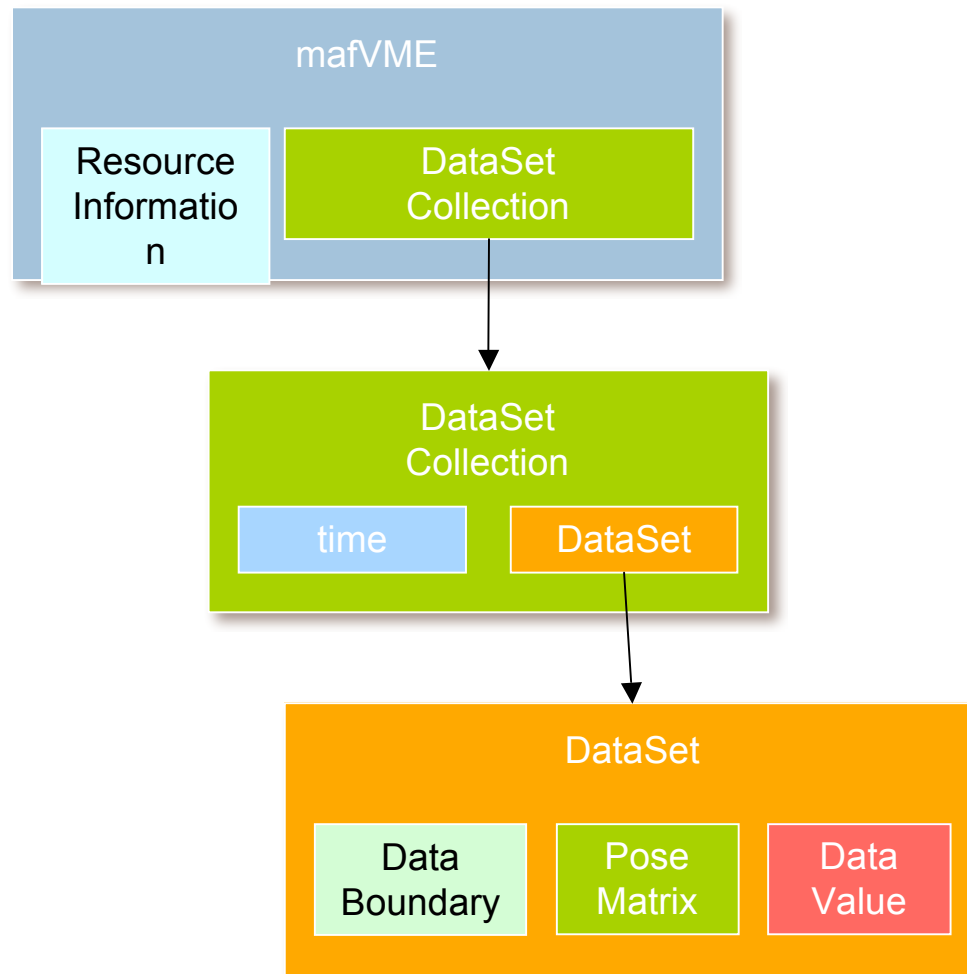


- Allows generating the system clock for animated data,
- Allows creating custom timers
- Provides a library of functions for the manipulation of time-space concepts
 - ▣ Change of reference system type
 - ▣ Pose matrices operations
 - ▣ Functions for the verification of topological conditions (i.e. rigid transformation)

MAF3 Resource



Data in MAF3



Resource Management Module



- Allows managing available resources by:
 - Add/Remove resources
 - Initialize a resource
 - Keep track of active resource
 - Garbage unused resources
 - Ask the resource to return a frozen state of its parameters to be stored using the memento pattern (can be used for undo)
- Define separation between object factory, resource managers, and component libraries (i.e. pipes)

Hierarchy Management Module



- Allows keeping resources organized in hierarchy
- Allows resources to be linked one each other (not only in a parent-child manner)

Logic Module

- Implements call-backs toward the selected GUI library
- Implements the logic used by the application

Plug-in Extensibility



- MAF3 resources can be extended through a plug-in mechanism.
- Data sharing can be
 - ▣ Through memory
 - ▣ Through a file
- Plug-in may contain pipes that are source-code compatible to the inclusion in core pipe lib

Scripting Language



- All modules can be wrapped through Swig into scripting languages like:
 - Python
 - Java
 - ...

Multithreading support



- Multithreading execution for MAF Operations or DataPipe algorithms will be available to the developers.

Internet References



- MAF3 collaboration area can be found at:
 - http://www.biomedtown.org/biomed_town/maf3



THANK YOU!