

WE LOOK AFTER THE EARTH BEAT



SMP2 Modelling using the K2 Simulation infrastructure

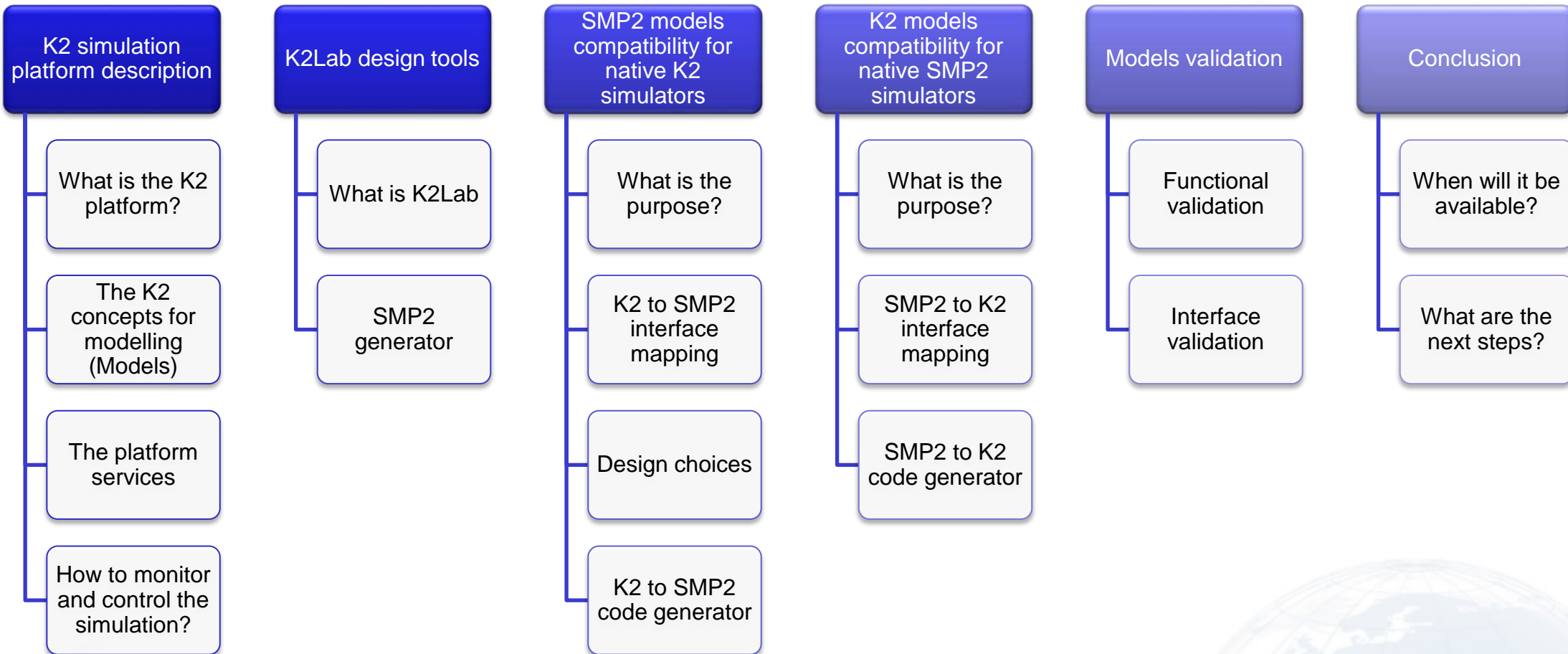
Laurent Cohen

laurent.cohen@thalesaleniaspace.com

26th March 2015

OPEN

Agenda



OPEN

What is the K2 platform

The K2 simulation platform is a proprietary software made by Thales Alenia Space

The Core K2 simulation platform is composed of :

- A set of libraries developed in C++, Python and Java to create models and simulators
- A set of third parties software
- Running on Linux

The K2Lab modelling tools

- Eclipse RCP graphical application
- Allow the user to create a modelling of K2 models and simulator
- Generates code, requirement and validation documents

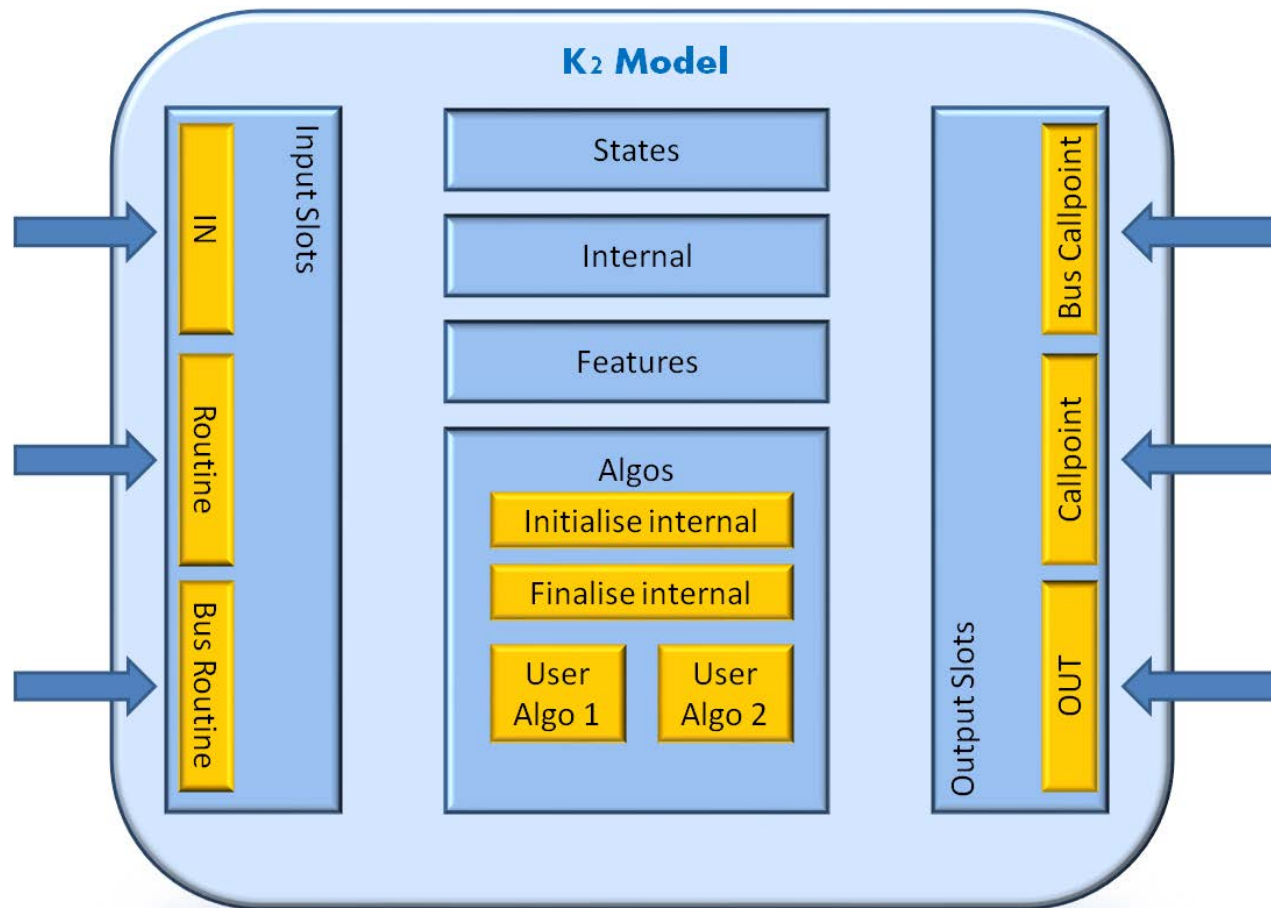
The eTestLab simulation execution command and control

- Graphical interfaces to configure, launch and stop simulations
- To control a simulation (change the speed, disconnect/reconnect models, send commands...)
- And to monitor (view models states, analyse the logs, ...)

OPEN

The K2 platform models

The K2 Model main concepts



The Input / Output slots : the only way to communicate

- Callpoint/Routine for asynchronous communication
- IN/OUT for synchronous communication

The model state is defined by

- Its real-time states
- Its internal data
- Its static features

The Algorithms are equivalent to SMP2 Entry Points

Each K2 model has a set of predefined algorithms (Initialise, Finalise internal, ...)

OPEN

The K2 platform services

The main services provided by the K2 platform

The K2 Time keeper

- Equivalent to the SMP2 time keeper

The K2 Scheduler

- Manages cyclic and acyclic events
- Schedules algo executions with date and priority
- Cyclic events are re-posted at the end of the execution
- Choice of the scheduler policy

The K2 Tracers

- Activation tracers to log the callpoint/routines calls
- Data tracers to log some of the model states periodically

The K2 Logger

- Log informations with a severity
- Can be retrieved in real time with a graphical application
- A profiler can be activated to log the time spent in each algos or routines

The Save/Restore feature

- Save automatically the simulation state
- Restore a saved context to resume the simulation

TM/TC service

- Send satellite TC
- Receive satellite TM in real-time

XMLRPC control

- The K2 simulator is self-contained and do not need any user interaction
- The K2 platform provides an XMLRPC server allowing external tools to monitor and control the simulation

OPEN

The K2 platform monitoring and control

The K2 platform provides graphical interfaces to monitor and control the models and simulations in real-time : eTestLab

Launching

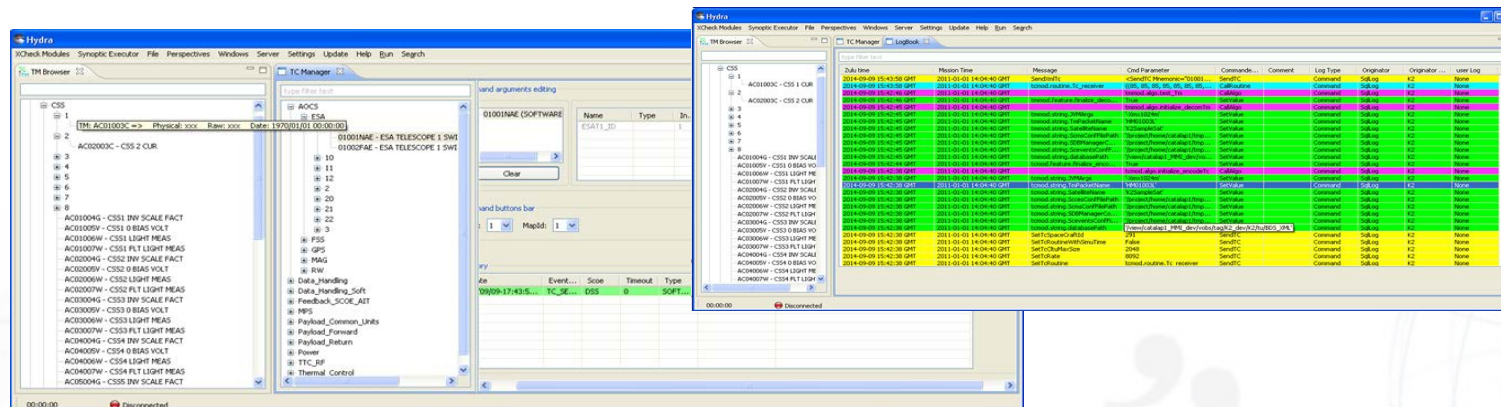
- Configure simulations
- Launch and stop simulations
- Manage existing simulators

Monitoring

- Models introspection
- Watch / Graph views
- Synoptic
- Log viewer and sorter

Control

- Change execution speed
- Disconnect / reconnect inputs and outputs
- Disconnect / reconnect callpoints and routines
- Change state values
- Call routines and callpoints
- Send commands



OPEN

The K2Lab design tool

The K2Lab graphical interface

- Eclipse RCP application allowing to
 - Design the K2 model interfaces
 - Develop the functional behaviour
 - Create simulators by instantiating and connecting models together
- Based on Eclipse Modelling Framework (EMF) and Acceleo

Model designer

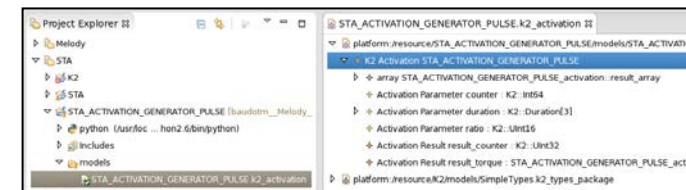
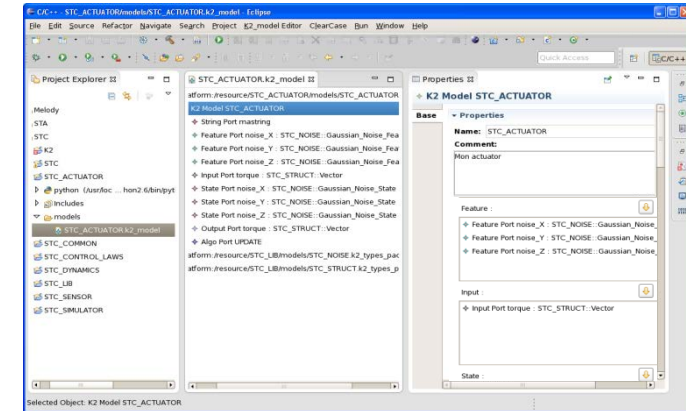
- Manage data of the same sort than the SMP2 catalogue file
- The user defines the interfaces of the model
- The internal data structures of the models (states, features)
- The algorithms of the models

Model generator

- Generates the K2 model skeleton from the design
- The eclipse C++ editors allow the user to fill the skeleton with the functionality
- The reverse engineering function

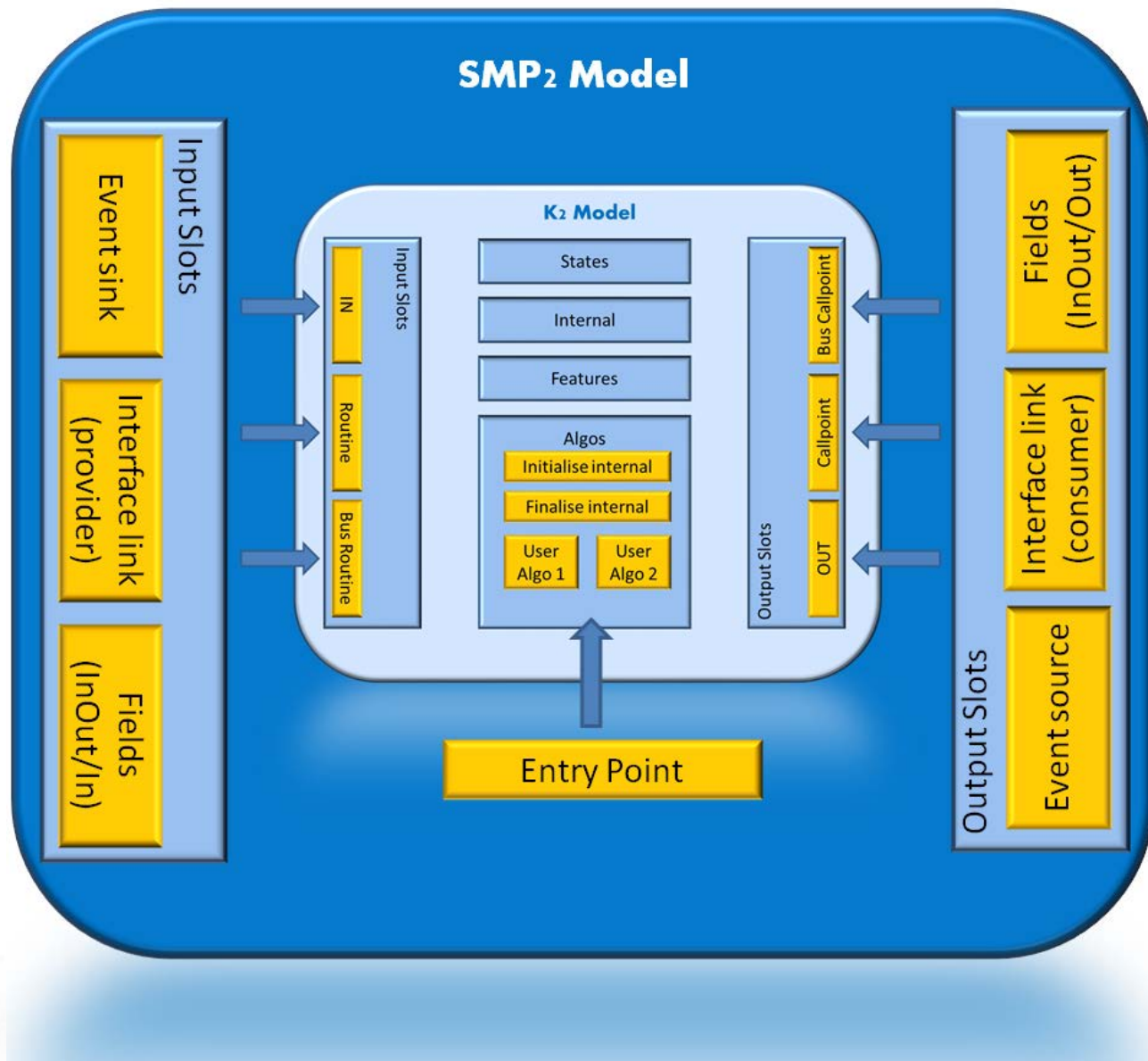
SMP2 Catalogue generator (planned)

- The K2 model is translated into an SMP2 catalogue
- An automatic translation of K2 concepts into SMP2 concepts is done



OPEN

SMP2 models compatibility for native K2 simulators



SMP2 types are fully compatible with K2 types

The connection concepts are different

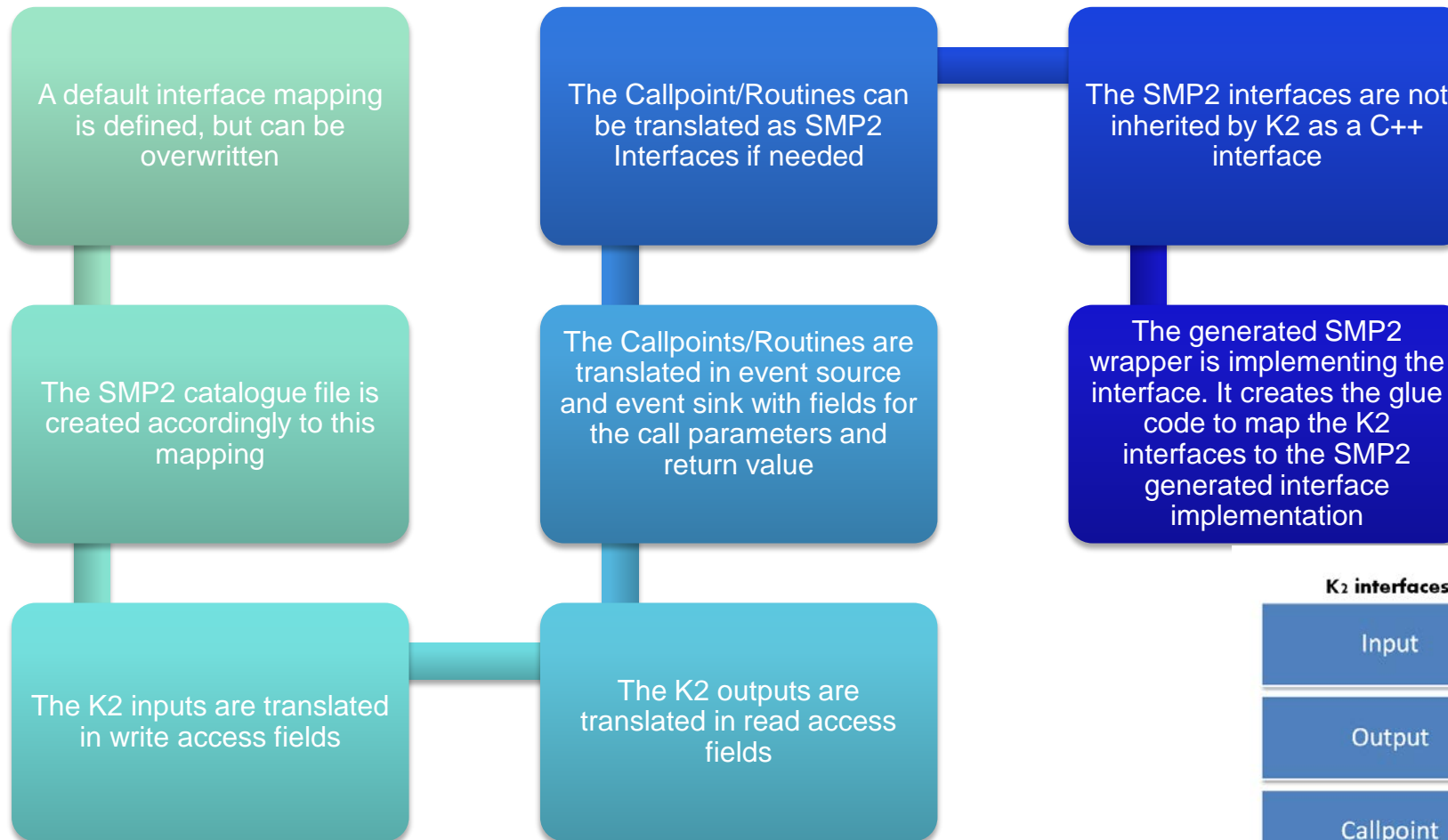
The operations to wrap a K2 model will be fully automated

The choices to translate a K2 concept into an SMP2 concept are forced by the catalogue generator

It is however possible to overwrite the predefined choices to select a more appropriate one when needed

OPEN

K2 to SMP2 interface mapping

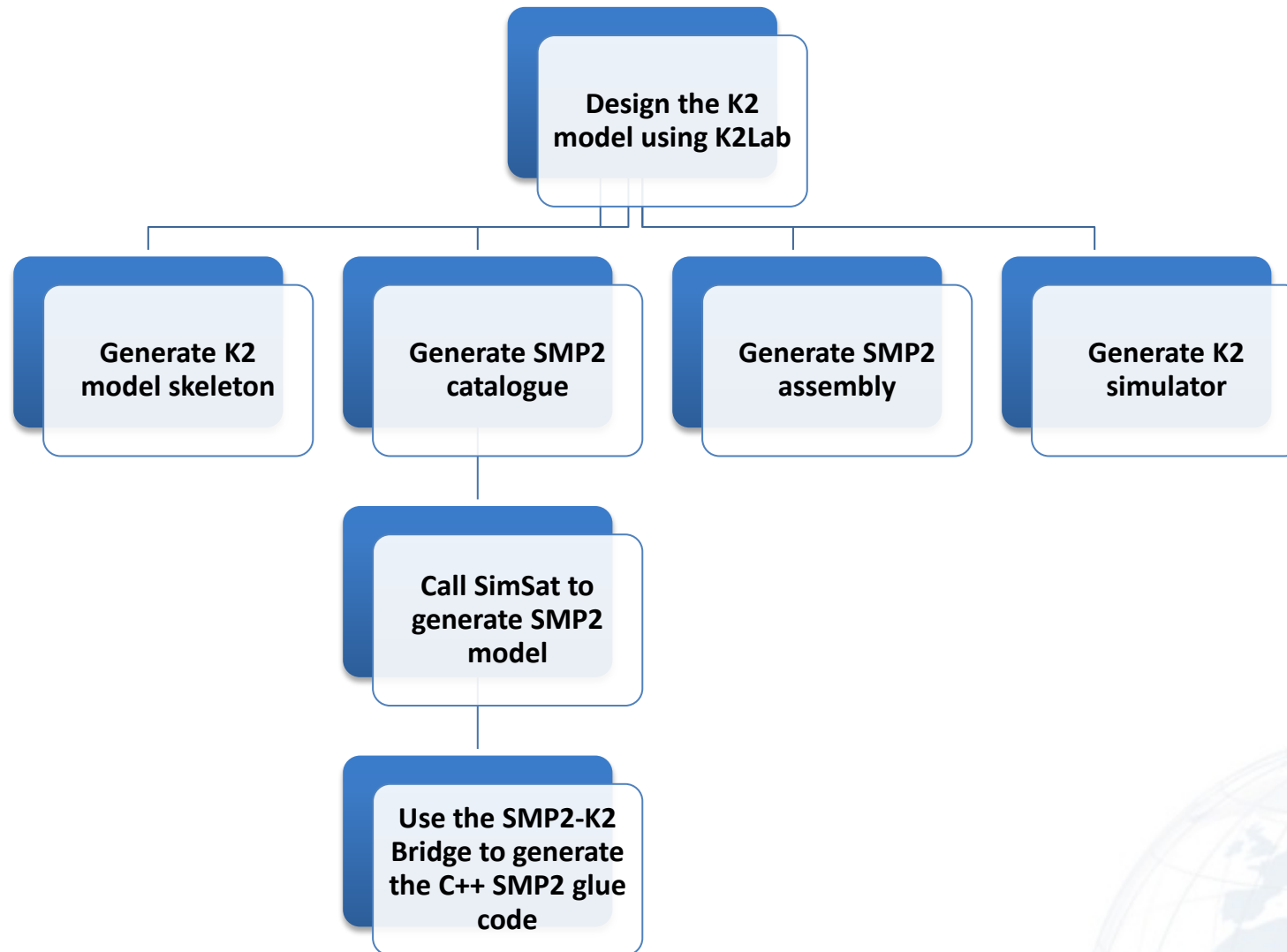


K2 interfaces	SMP2 interfaces
Input	Field (write access)
Output	Field (read access)
Callpoint	Event Source and Fields
Routine	Event Sink and Fields

Default mapping, configurable

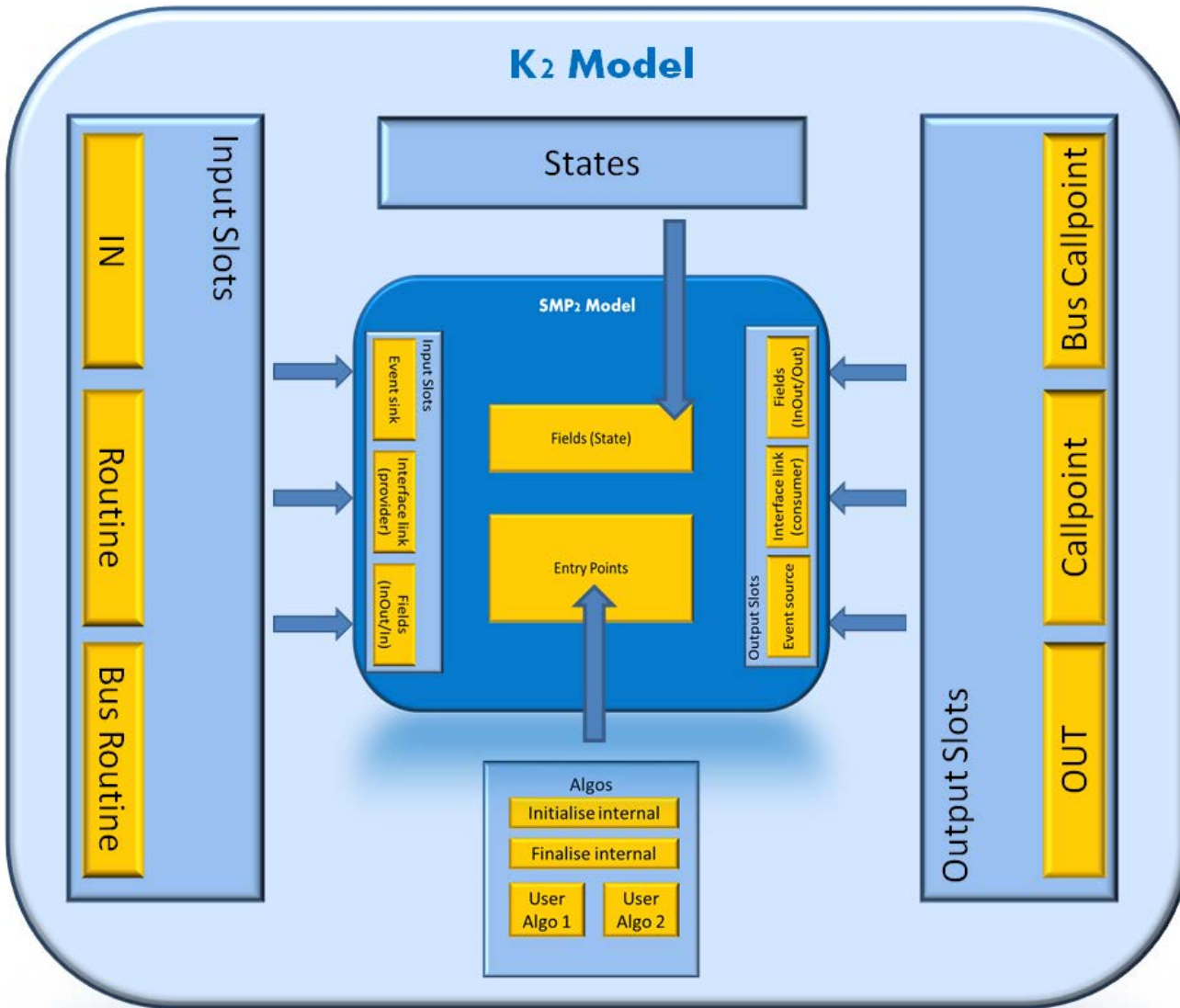
OPEN

K2 to SMP2 code generator with K2Lab



OPEN

K2 models compatibility for native SMP2 simulators



The SMP2 catalogue is the entry point to generate the K2 wrapper

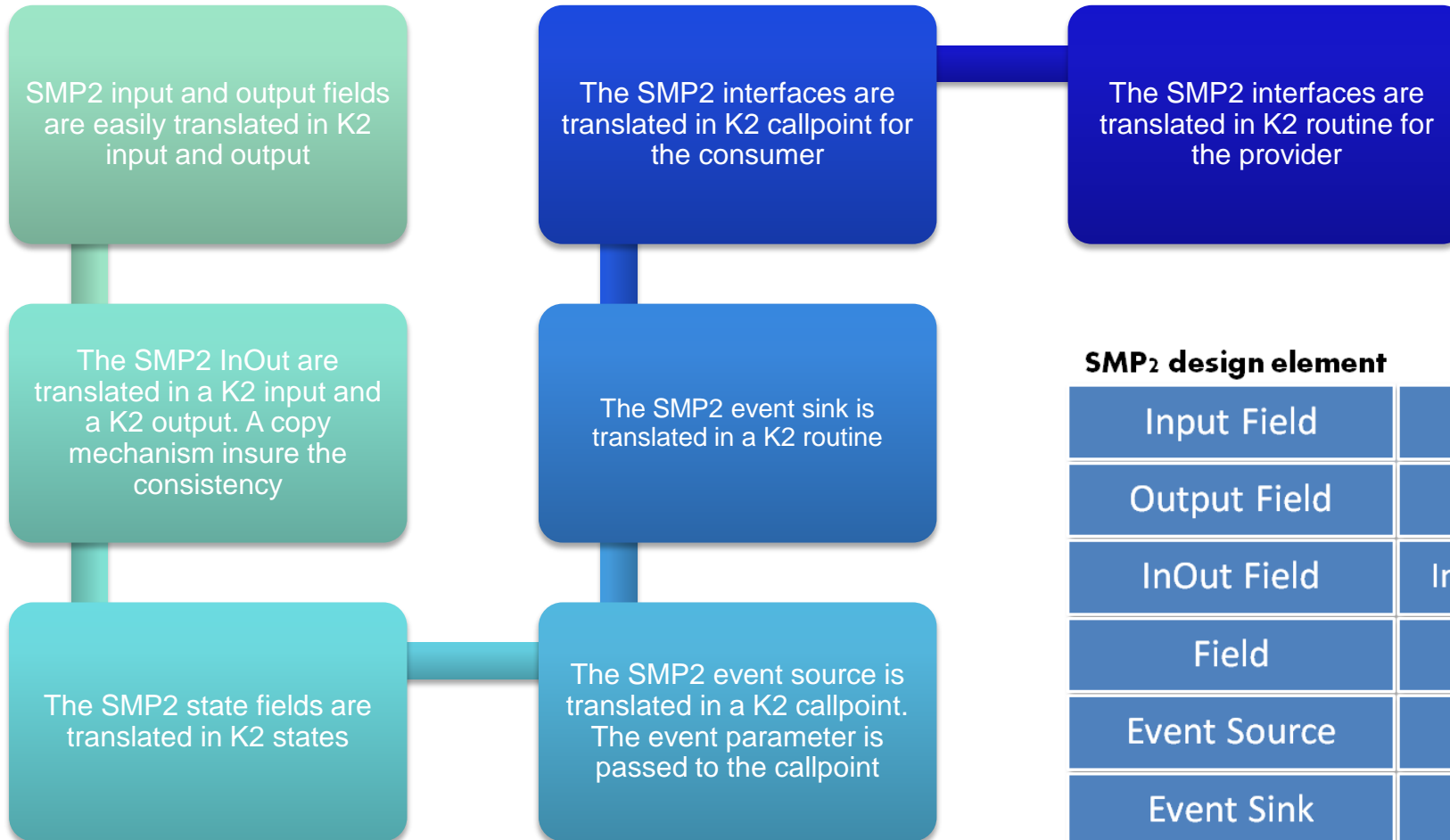
The operations to wrap an SMP2 model will be fully automated

The choices to translate an SMP2 concept into a K2 concept are forced by the K2 wrapper generator

It is however possible to overwrite the predefined choices to select a more appropriate one when needed

OPEN

SMP2 to K2 interface mapping

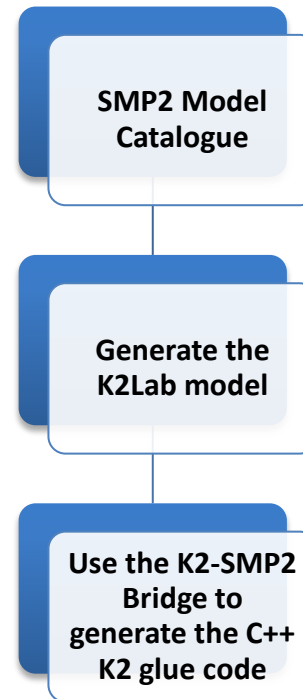


SMP2 design element	K2 interfaces
Input Field	Input
Output Field	Output
InOut Field	Input and Output
Field	State
Event Source	Callpoint
Event Sink	Routine
Interface	Callpoint/Routine

Default mapping, fixed

OPEN

SMP2 to K2 code generator with K2Lab



OPEN

© 2015, Thales Alenia Space

Models validation

The SMP2 models validation with BASILES are prepared in TCL

The K2 models validation with the K2 platform are prepared in Python

The automatic test conversion is not obvious

SMP2 native models

Translate tests manually from TCL to Python (TCL test is the specification)

K2 native models

Translate tests manually from Python to TCL (Python test is the specification)

OR

SMP2 native models

Functional validation using SMP2 platform

Interface validation using K2 platform

K2 native models

Functional validation using K2 platform

Interface validation using SMP2 platform

OPEN

Conclusion

The solutions presented will be fully automated and operational on Q3 2015

This will open up new opportunities to share and reuse existing models

This approach is seen as the first step to have a K2 platform natively compatible with the SMP2 standard

OPEN

© 2015, Thales Alenia Space

Questions ?

Thank you for your attention



Laurent Cohen

Laurent.cohen@thalesaleniaspace.com

OPEN

© 2015, Thales Alenia Space

ThalesAlenia
Space
A Thales / Finmeccanica Company