Complex Systems \ EB RACE

Analyzed ▸ Understood ▸ Improved ▸ Automated

Elektrobit

Torsten Mosis

December 3, 2015

Agenda

- Challenges in IVI SW Architecture
- EB RACE
  - Key Features
  - Architecture
  - Customization and Extensibility
  - Automation and Validation
- Summary

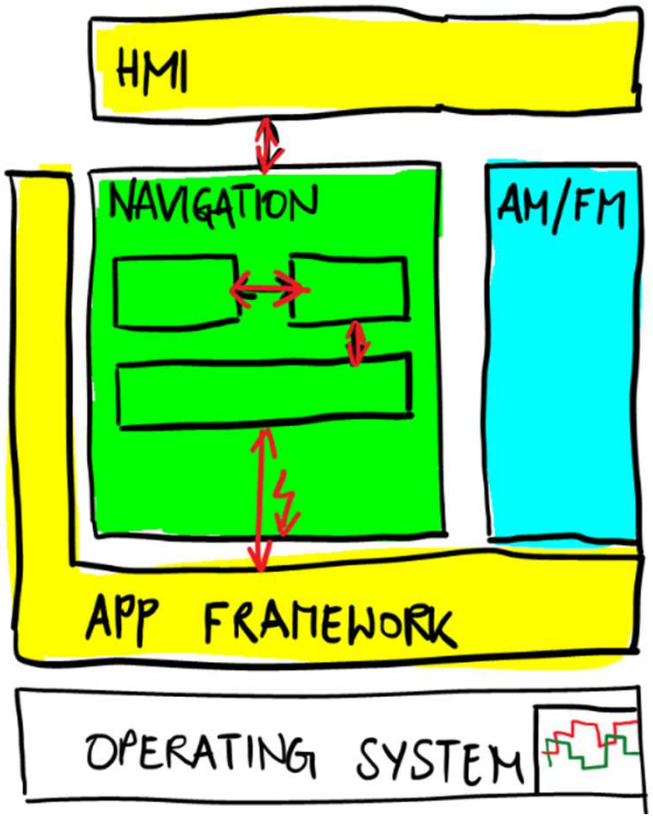
© Elektrobit (EB) 2015 |

Architecture of an Infotainment System is complex

Features are cross-cutting several SW Layers

Different SW Layers are implemented by different suppliers

The same may apply for the applications

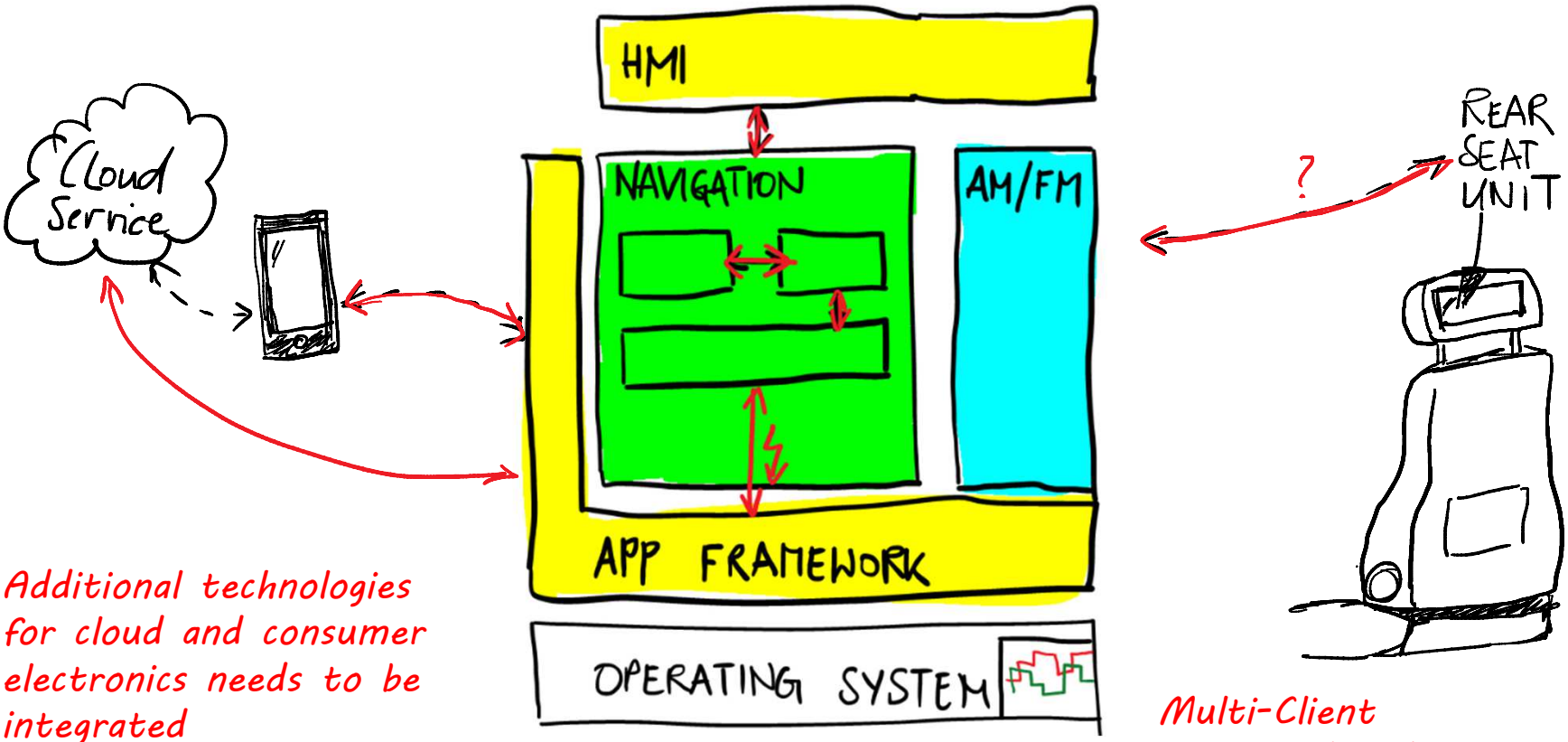


Various technologies are used for communication between SW components

Resources must be shared between different applications

© Elektrobit (EB) 2015

… and becomes even more complex in future

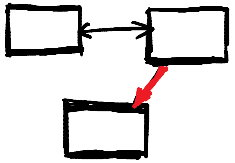


Additional technologies for cloud and consumer electronics needs to be integrated

Multi-Client functionality becomes more relevant

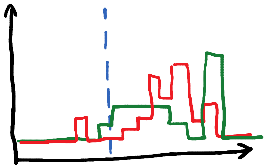
© Elektrobit (EB) 2015

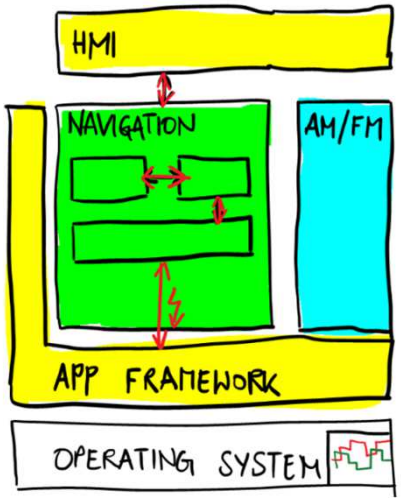
What do you want to know about your system?

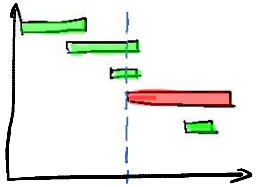


How do your components interact with each other?

What about the workload and balance of your system?

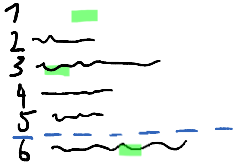




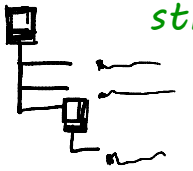


Does your software behave as expected?

Are there hints for malfunctions in your logs?

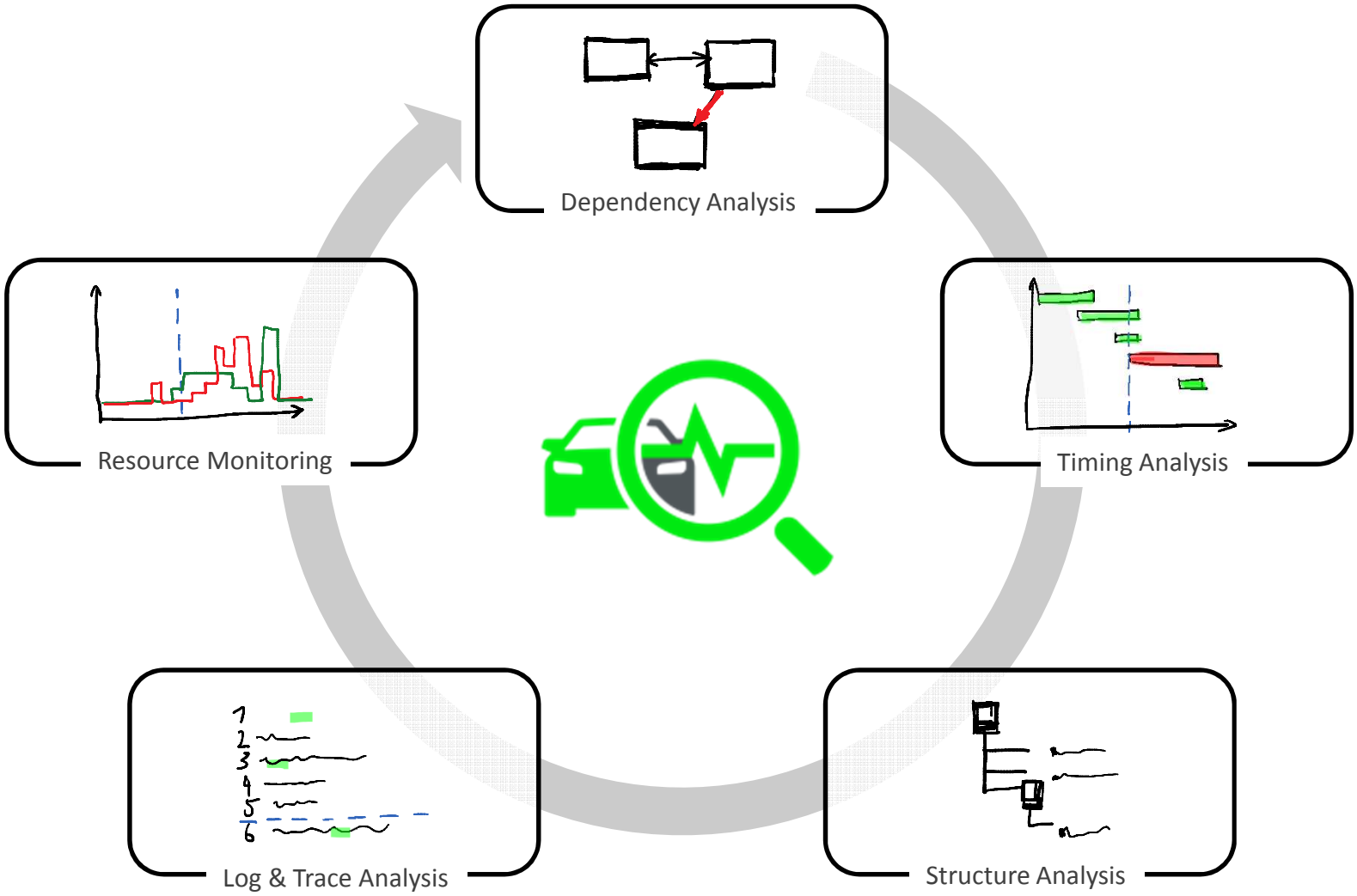


How is your system structured?



© Elektrobit (EB) 2015

Acquire, Aggregate & Correlate Data with EB RACE



© Elektrobit (EB) 2015

EB RACE is…

… a development tool for **collecting, aggregating and correlating data** and its visualization.
Live and post-mortem.

… a method to **identify and localize functional and non-functional defects** in an early phase for reducing the efforts spent for testing and bug fixing in later phases.

… a sophisticated approach for analyzing and validating **complex automotive software systems** from top to bottom.

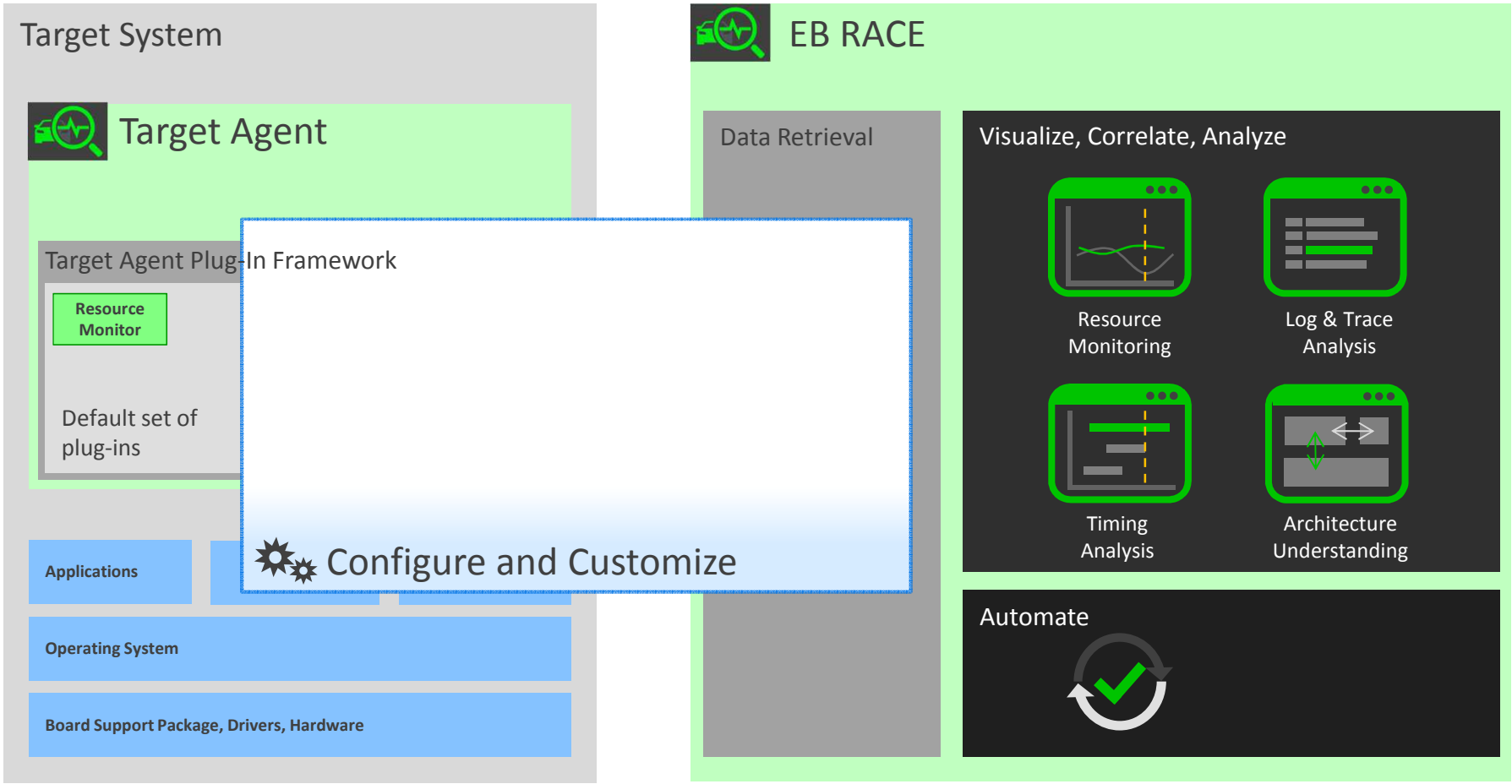
© Elektrobit (EB) 2015

The Key Features

- Graphical correlation and analysis of target runtime data
- Hot spot analysis & root cause isolation
- Built-in script engine based on Xtend
- Customizable data decoders
- Multi-OS target-agent with plug-in architecture
- Integration into continuous build and test toolchain with **EB RACE Auto** mode

© Elektrobit (EB) 2015

Highly Adaptable Architecture

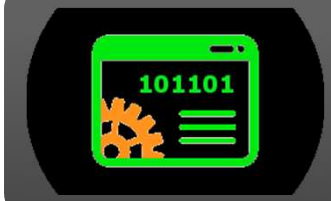


© Elektrobit (EB) 2015

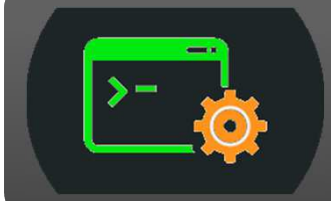
Extend and Customize EB RACE

Implement your own **Target Agent** Plug-In, that retrieves project/system specific data

Provide your own **Importer** for any kind of logs, which are not created originally by the target agent



Hook in your own **Decoder**, that transforms arbitrary non-primitive data (e.g. binary payload) into structured readable text

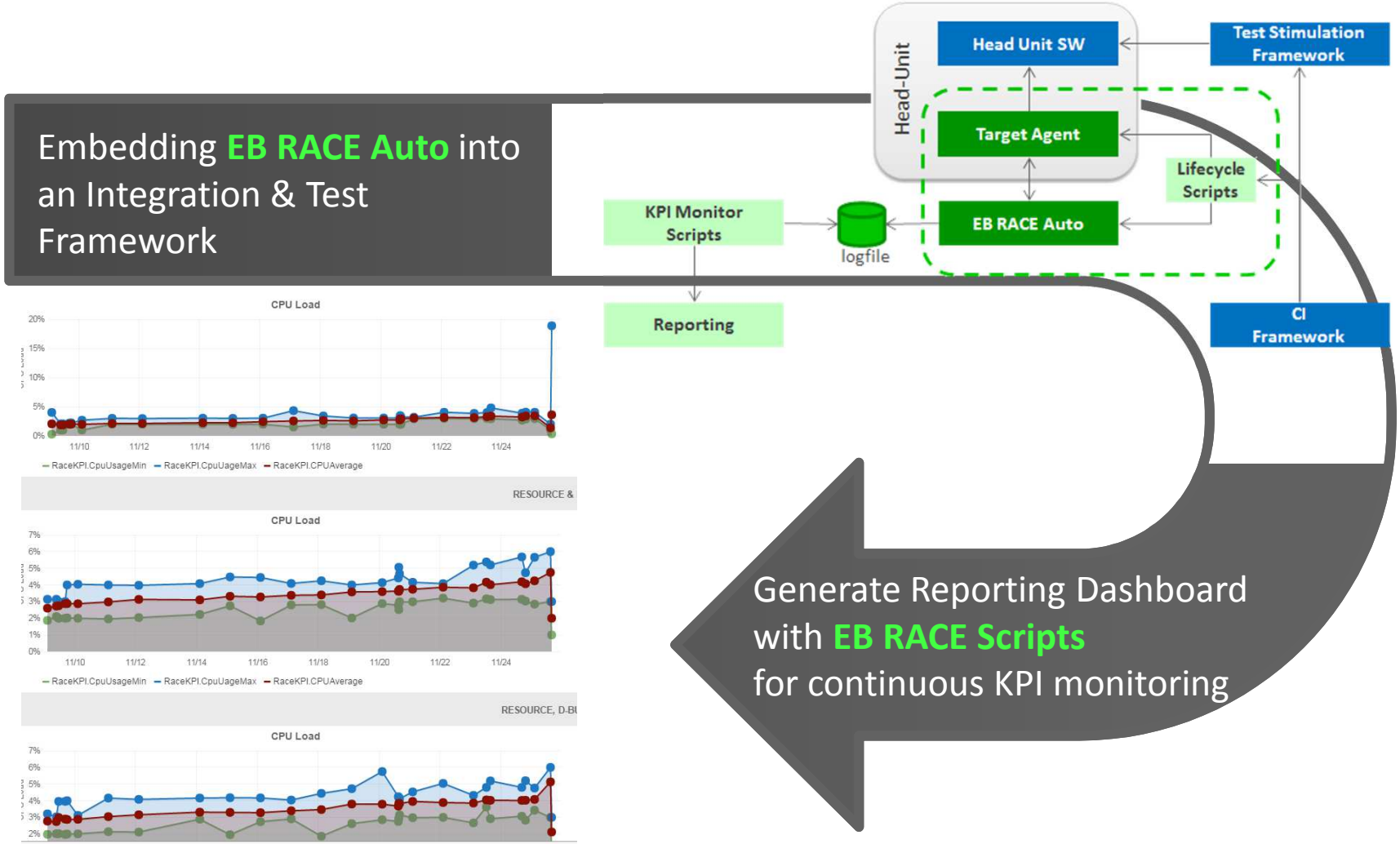


Use the Built-In **Script** Engine to add new functionality by accessing the EB RACE raw data and resources

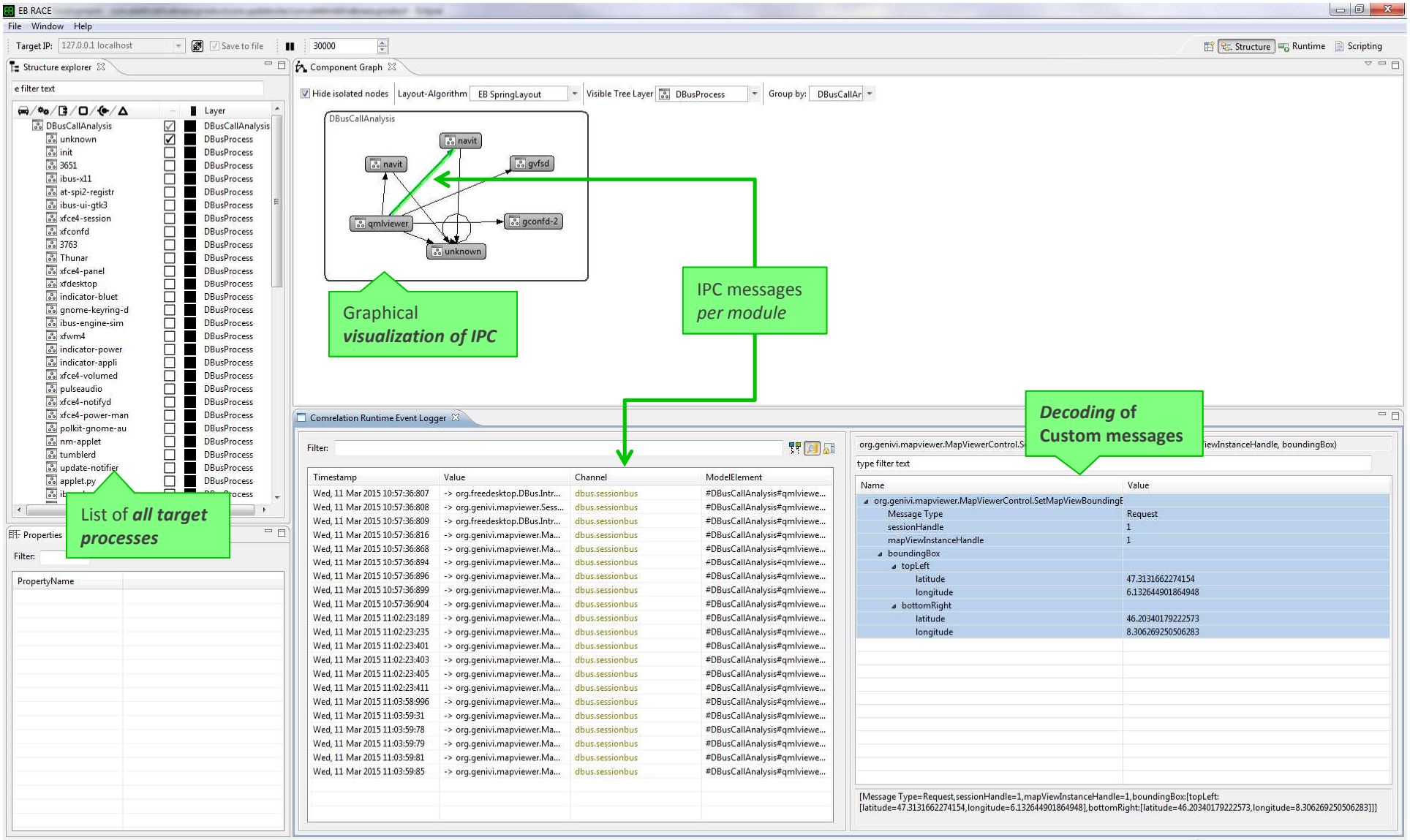
© Elektrobit (EB) 2015

EB RACE Auto

Embedding **EB RACE Auto** into an Integration & Test Framework



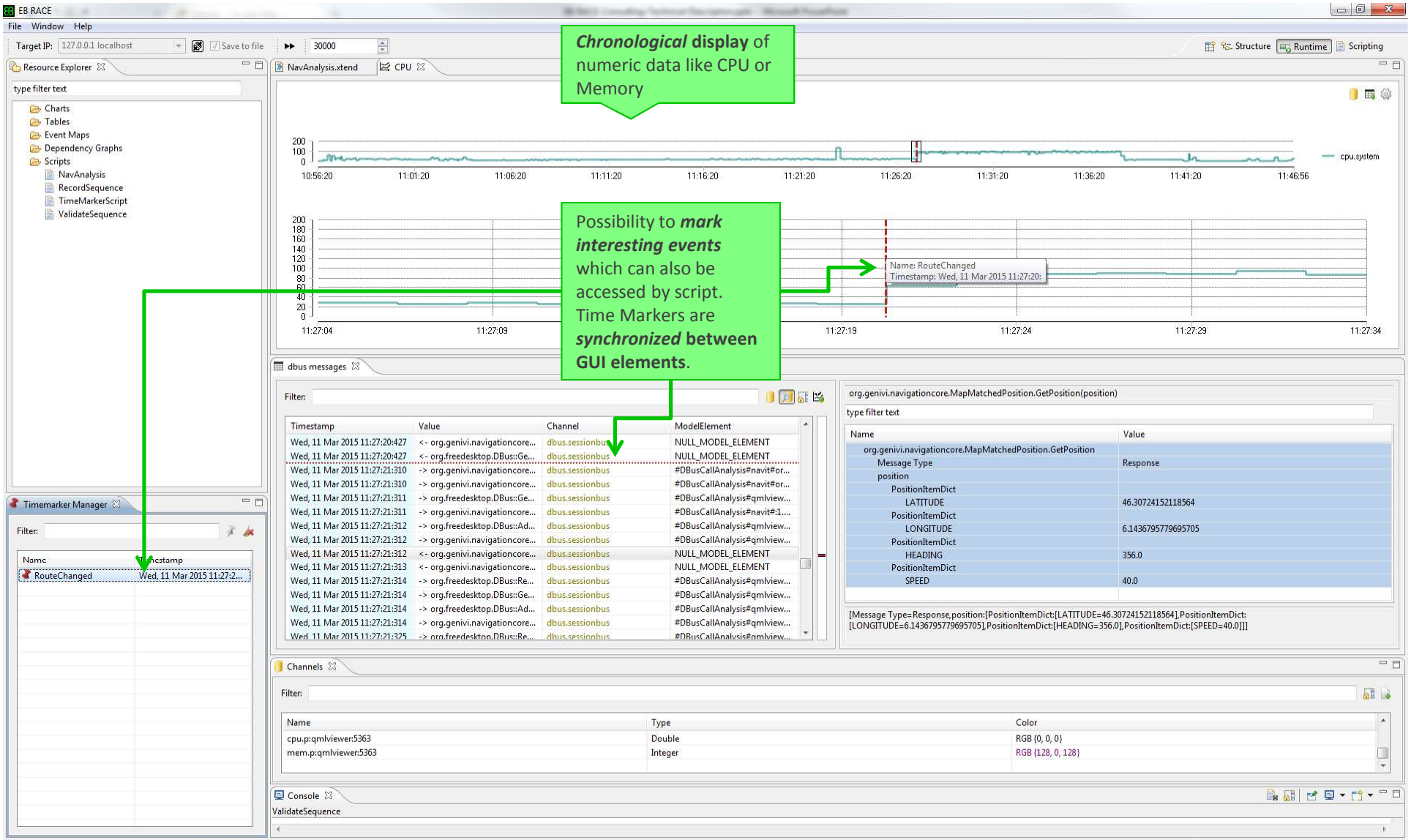
© Elektrobit (EB) 2015



© Elektrobit (EB) 2015

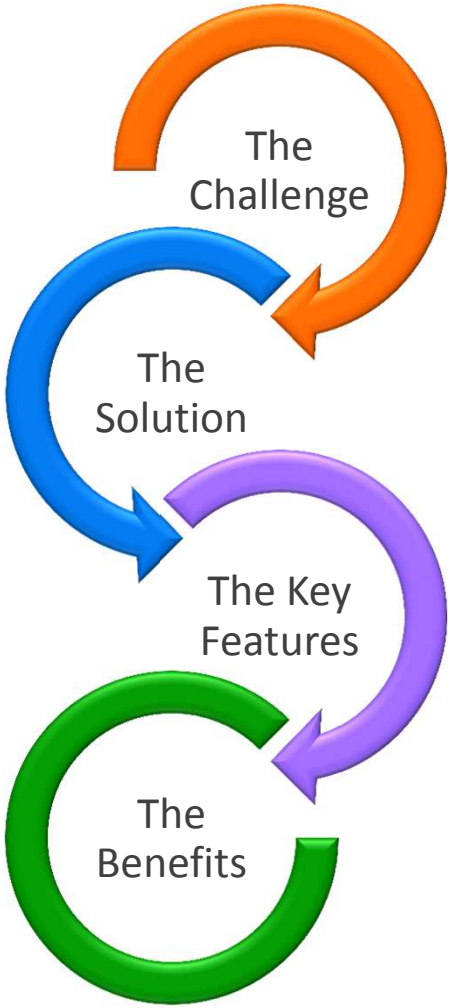
EB RACE

Elektrobit



© Elektrobit (EB) 2015

EB Race Summary



- Comprehend complex software systems
- Track down functional and non-functional defects
- Monitor stability and robustness

- EB RACE to inspect, analyze and validate the software system continuously
- Experts to customize and embed EB RACE into the project eco-system

- Graphical correlation and analysis of target runtime data
- Hot spot analysis / root cause isolation
- Integration into continuous build and test toolchain with EB RACE Auto mode

- Assess system health
- Evaluate specification compliance
- Improve quality and reduce development cost

© Elektrobit (EB) 2015

Please visit us at the Demo Room and Workshop:

DEMO

Montcalm Room
09:00 am – 04:00 pm

EB RACE Hands On Demonstration

WORKSHOP

Nicolet Room
13:30 am – 04:00 pm

Model-based Development of Multimodal User Interfaces Including Speech Dialog

© Elektrobit (EB) 2015

Contact us!

Elektrobit

sebastian.koenig@elektrobit.com
torsten.mosis@elektrobit.com

www.elektrobit.com/products/software-engineering/software-integration-and-engineering-services/eb-race/