

# TCF

## Anyware's Implementation of the Target Communication Framework

What is it ?



# Why a new TCF implementation ?

- We work on an IDE that require communications with targets.
- We need the TCF model :
  - ▶ notions of services.
  - ▶ command and event messages.
  - ▶ fully asynchronous.
  - ▶ multiplexing over a single communication channel.
  - ▶ value-add servers possibilities.
- However the context is specific :
  - ▶ impossible to implement an agent on the target.
  - ▶ a set of custom services.
  - ▶ a custom binary communication protocol.
  - ▶ a custom communication channel (serial port and specific driver).
  - ▶ services may evolved and need to be replaced with Eclipse mechanism (plug-in and extension point).



# What we needed to modify

- De-couple the notion of communication channel from the message management in order to simplify the add of new communication means (TCP, serial port, ...) and communication protocols (Jason, binary, custom, ...).
- De-couple the notions of services (general identification), proxy (send commands and listen to events), provider (reply to commands and send events), value adder (intercept commands and events).
- Delegate the management of these notions (instantiating, publication, subscription, ...) to external managers.
- Possibility to categorised all this notions with different types.
- Possibility to contribute to this notions through extension points.
- Have a unique and thread-safe access for framework users

# What is not yet implemented

---

- The Locator service (available services, redirection, ...).
- The notion of Context Identifier.
- The management of congestion level.
- Integration of existing TCF Services.
- Full compatibility with the C agent.



- A centralised error management with extensible error handler policies.
- A centralised log facility with extensible logging possibility.
- An extensible discovery management.
- A user/developer help.
- May be an encapsulation in ECF.
- May be connectors to messaging and presence protocols (ActiveMQ, XMPP, ...).
- The possibility to use EMF Transaction and Data Binding.



# How does it works ?



- Only one access point the “TCF” class.
- It enables through thread-safe static methods :
  - ▶ Add/Remove Peers
  - ▶ Connect/Disconnect Peers
  - ▶ Retrieve services proxies
  - ▶ Add/Remove listeners on Peer State

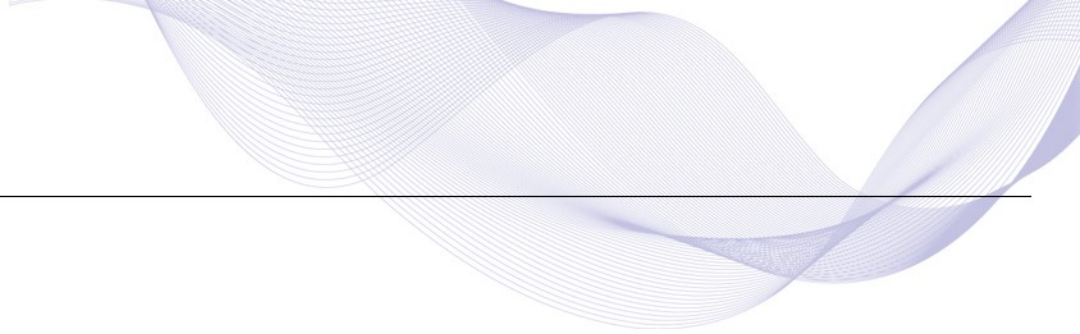


- Developer can add its own types of the following elements:
  - ▶ Peers.
  - ▶ Communication Channels.
  - ▶ Services Proxies.
  - ▶ Services Providers.
  - ▶ Services Value-Adders.
- Each types is identify by a unique string.
- The contribution are made through an specific extension point
- For each contributed element, the developer must provide a manager class that implement a specific interface.
- TCF delegate the elements life-cycle management of an element (creation, destruction, availability, ...) to its manager.
- TCF provide facilities to help the manager (send/listen command and event, services availability, ...)

- Message abstraction (type, token, service, name, data) for Event, Command, Result and Progress.
- The Communication Manager manage communications life-cycle with peers thanks to Peer Communication instances.
- Peer Communication manage :
  - ▶ Services (Proxies and Providers) and Communication Channel by calling the right managers.
  - ▶ Messages routing between these entities.
  - ▶ If Service Value-Adder are needed, a Value-Adder Communication manage it by intercepting messages between Peer and Peer Communication.
- The Task Manager manage the threaded event queue (runnables) used for asynchrony.
- The TCF Manager manage the framework initialisation by instantiating all the managers (internal and provided).



# What kind of contribution ?



- Anyware is very interested in TCF development and wishes to join the project.
- Anyway, the code can be made available under EPL (with the associated application example).
- It (or parts of it) could be used for the future TCF, if developers find it useful.
- Feel free to ask questions :
  - ▶ mail : “[gaetan.morice@anyware-tech.com](mailto:gaetan.morice@anyware-tech.com)”.
  - ▶ Gtalk : “[gaetan.morice@gmail.com](mailto:gaetan.morice@gmail.com)”.



**anyware**  
TECHNOLOGIES

A **wavecom** COMPANY

Linking Innovation to your Business

**THANKS!**

