

XXII Jornadas de Tiempo Real

Desarrollo de una plataforma de
ejecución para validación y
verificación de código autogenerado.

Jesús Zurera, Hugo Valente, Miguel A.
de Miguel, Ángel G. Pérez, Alejandro
Alonso, Juan Zamorano, Juan A. de la
Puede.

STRAST



UNIVERSIDAD
POLITÉCNICA
DE MADRID



Dirección General de
Investigación e Innovación Tecnológica
CONSEJERÍA DE CIENCIA,
UNIVERSIDADES E INNOVACIÓN

Comunidad de Madrid



Introducción.

Objetivo. Generación autónoma de código fiable y validación basada en modelos de componentes.

- H1 Análisis y selección de lenguajes de modelado y generación autónoma de código.
- H2 Caracterización de un subsistema del UPMSat-2 basado en modelos.
- H3 Generación automática de código para los modelos de los subsistemas.
- H4 Análisis temporal e integración en plataforma de ejecución.
- H5 Validación Hardware-In-the-Loop.

Requisitos para selección un lenguaje de modelado.

Análisis de tendencias actuales en arquitecturas *software* basadas en componentes

- **core Flight System.**
- **AUTOSAR.**
- SAVOIR.
- Otras soluciones de código abierto y con gestor de mensajes: RabbitMQ, REST, GMSEC API.

Requisitos para el diseño del modelo de componente

Análisis detallado de cFS y AUTOSAR para la generación de un **modelo de componente**.

Abstracción de la plataforma para el desarrollo de requisitos. Especial atención al **sistema de comunicación -> interfaces**.

Integración con UI

Requisitos diseñados con objetivo de su **integración en una *User Interface*** para la representación de los subsistemas modelados.

Requisitos para un lenguaje de modelado



Requisitos para un lenguaje de modelado

N to M
Asynchronous
Communication

Fault
Detection

Data Store

Component
Management

Events

General
Requirements

Elección de un lenguaje
de modelado:

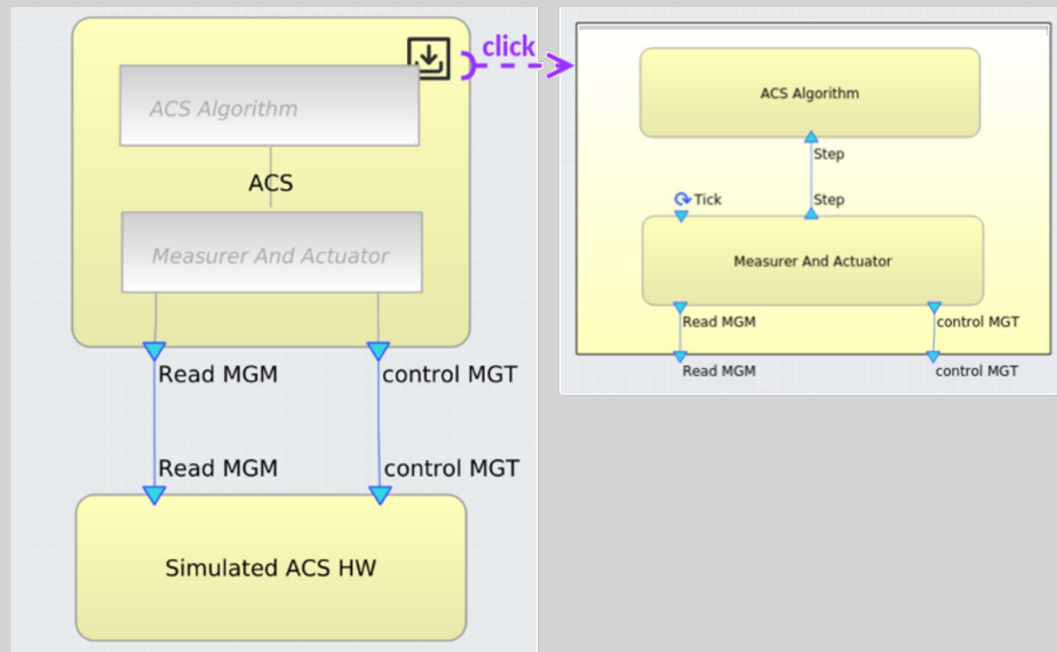
- AADL. Integrado en TASTE, herramienta de la ESA para desarrollo basado en modelos.

Elección herramientas de
generación de código:

- AADL Converter
- Kazoo
- Ocarina



Caracterización de un modelo de UPMSat-2.



Herramienta de modelado

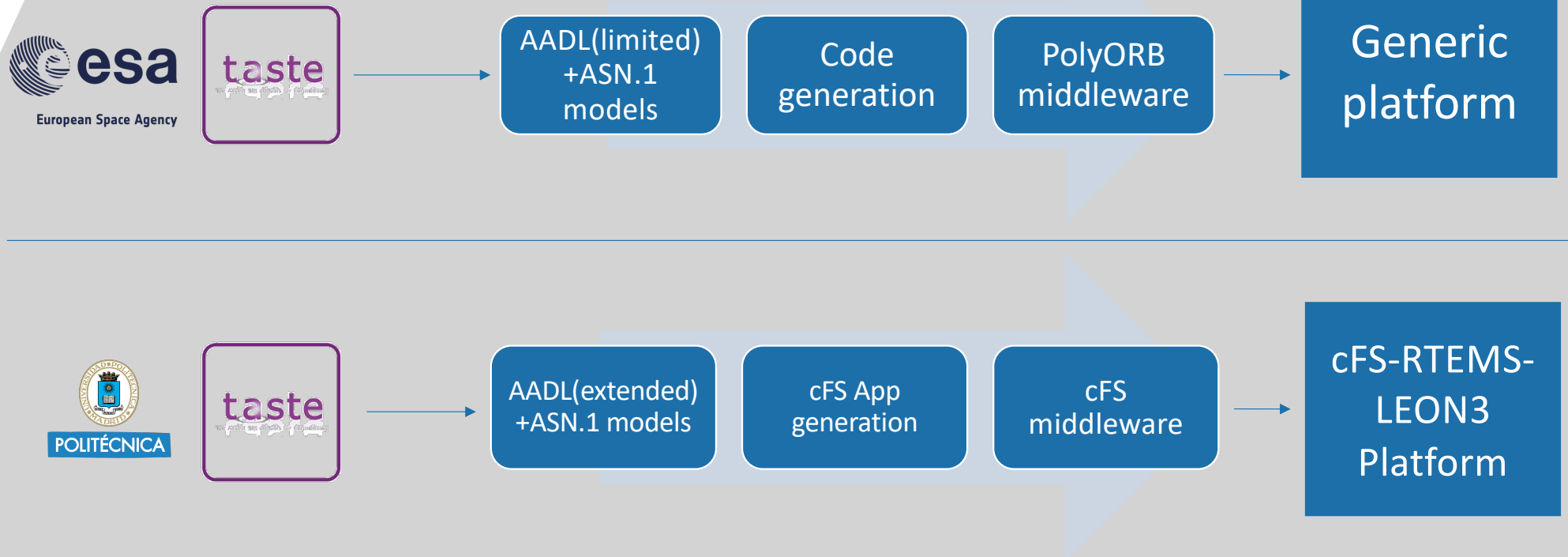
- Space Creator. Nueva versión de TASTE

Estructuras a modelar:

- ACS Algorithm QGenC generated
- Measurer and Actuator
- Simulated ACS HW

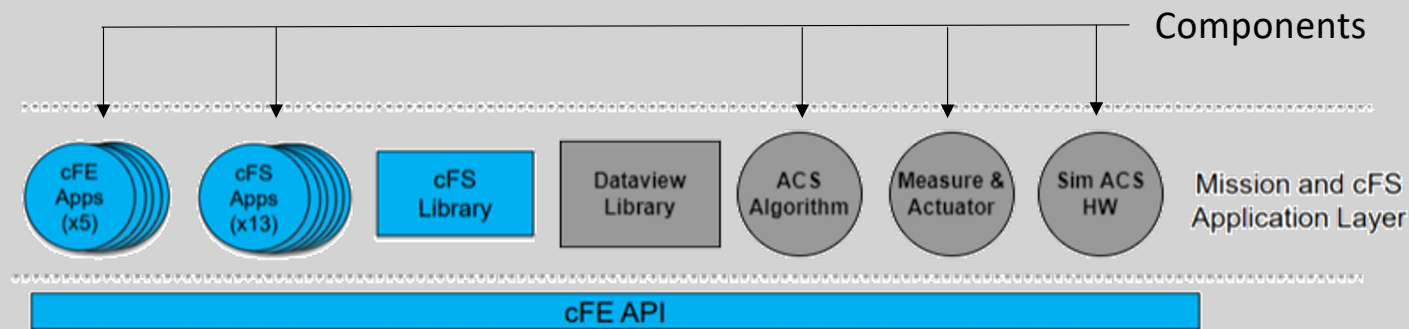


Esquema de modelado, generación de código e integración.

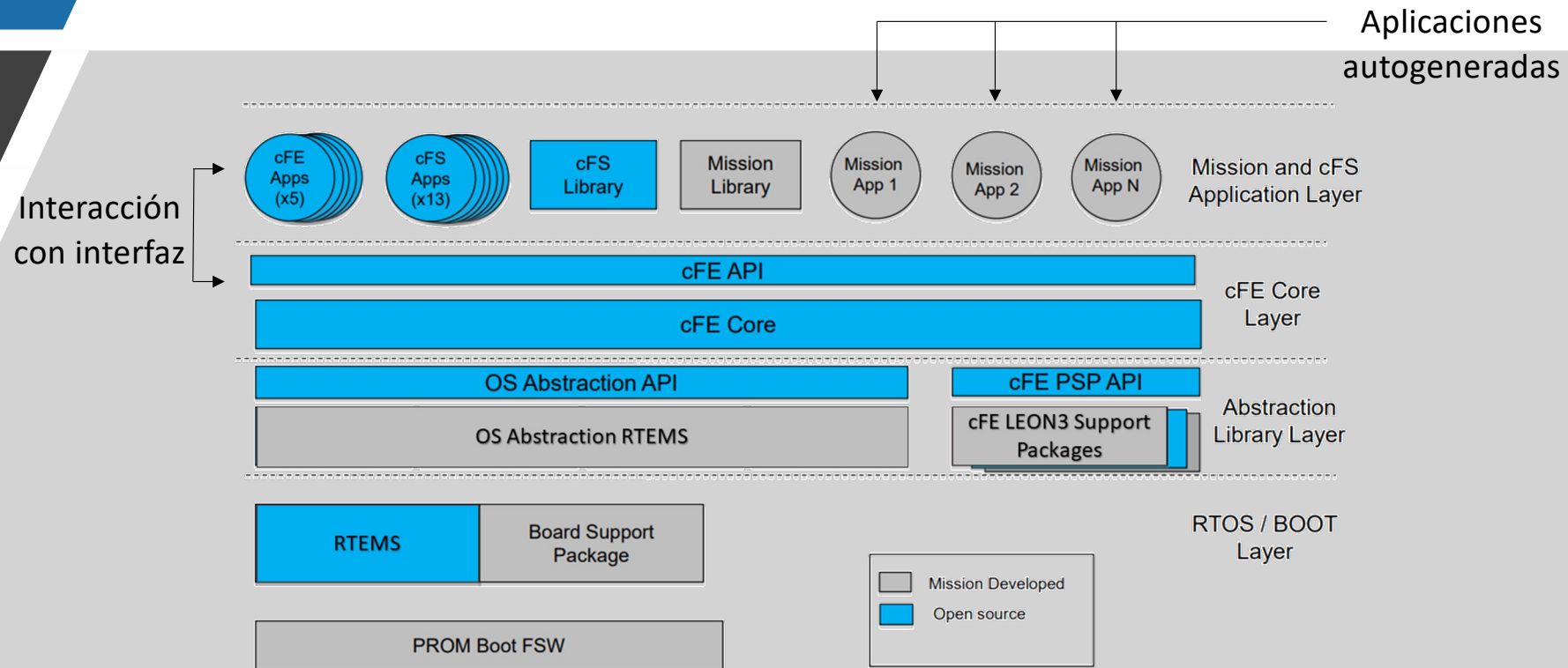


Modelo de componentes.

- El objetivo es diseñar un conjunto de componentes reutilizables, escalables y configurables que encapsulen las funcionalidades de cada modelo en una aplicación de cFS.
- ¿Cómo? Estandarizando cada bloque funcional, como aplicaciones de cFS. Cada modelo de los sistemas caracterizados en AADL transformado en una app de cFS.



Objetivo. Arquitectura cFS/RTEMS/LEON3.



Desarrollo. cFE Core Services.

core Flight Executive Services

- ES:** Administra el sistema y crea un entorno de ejecución de aplicaciones.
- EVS:** Proporciona un servicio para enviar, filtrar y registrar mensajes de eventos.
- SB:** Proporciona un servicio de mensajería de publicación/suscripción para aplicaciones.
- TBL:** Administra estructuras de datos en forma de tabla asociadas a las aplicaciones.
- TIME:** Gestión del tiempo en el sistema.

Executive
Services
ES

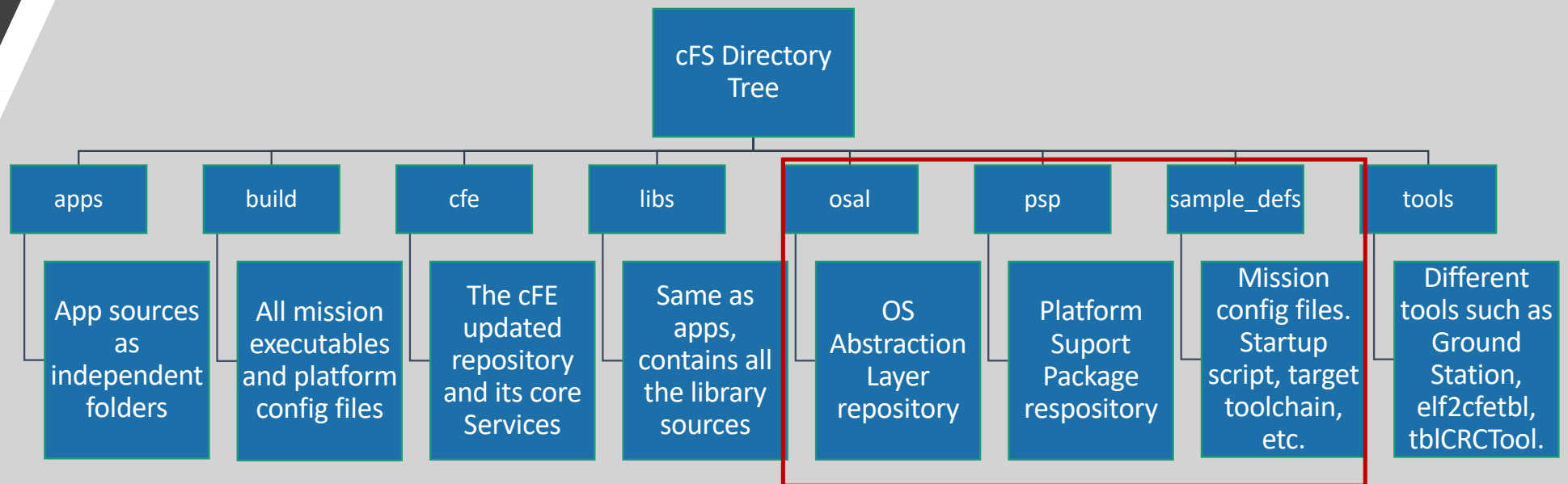
Event
Services
EVS

Software
Bus
SB

Table
Services
TBL

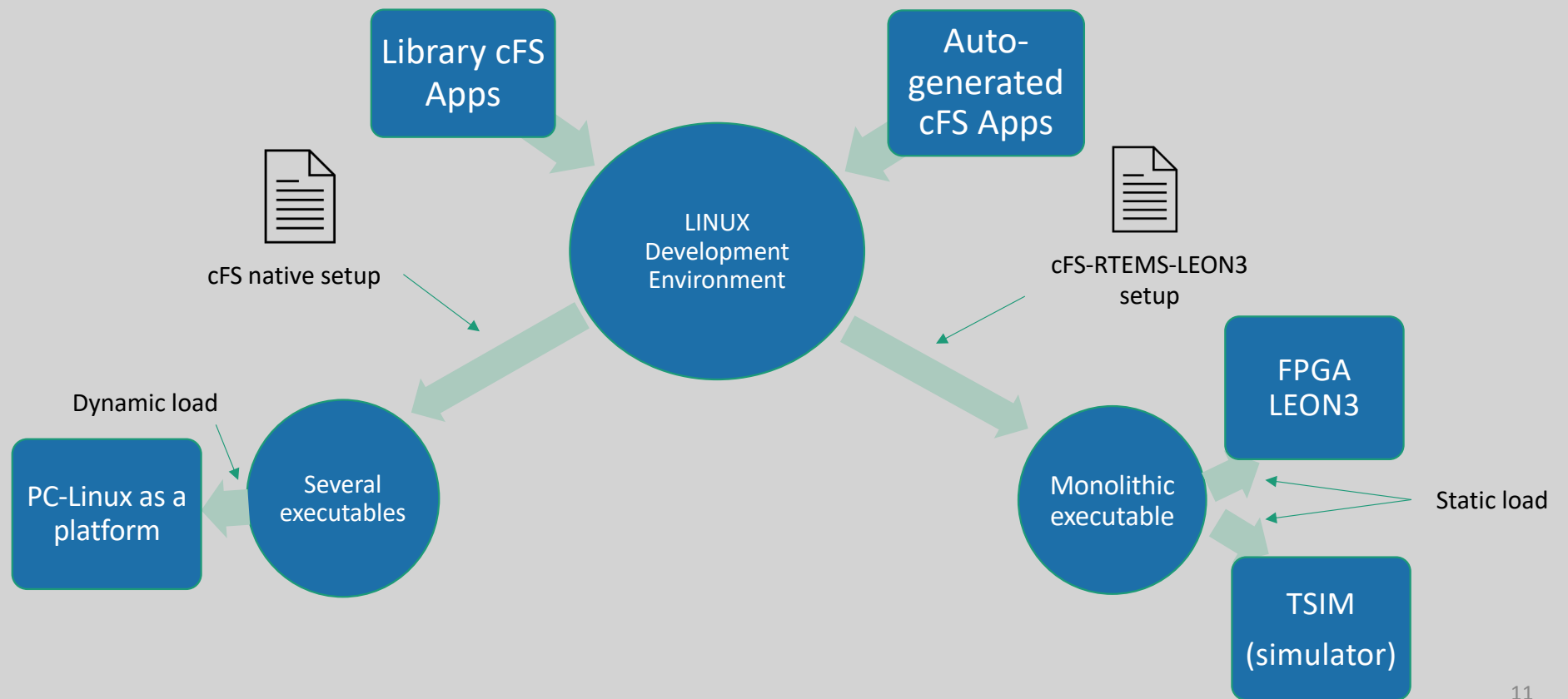
Time
Services
TIME

Nueva plataforma cFS. LEON3-RTEMS.



Due to cFS layered structure only changes needed are located here

Entorno de desarrollo.



Resultados. Testeo de la plataforma.

