

SAM – SISTEMA DE AUTOMATIZACIÓN DE MEDICIONES MODULAR Y EXTENSIBLE BASADO EN *PLUGINS*

Gastón E. Pérez⁽¹⁾, Pablo De Césare⁽¹⁾

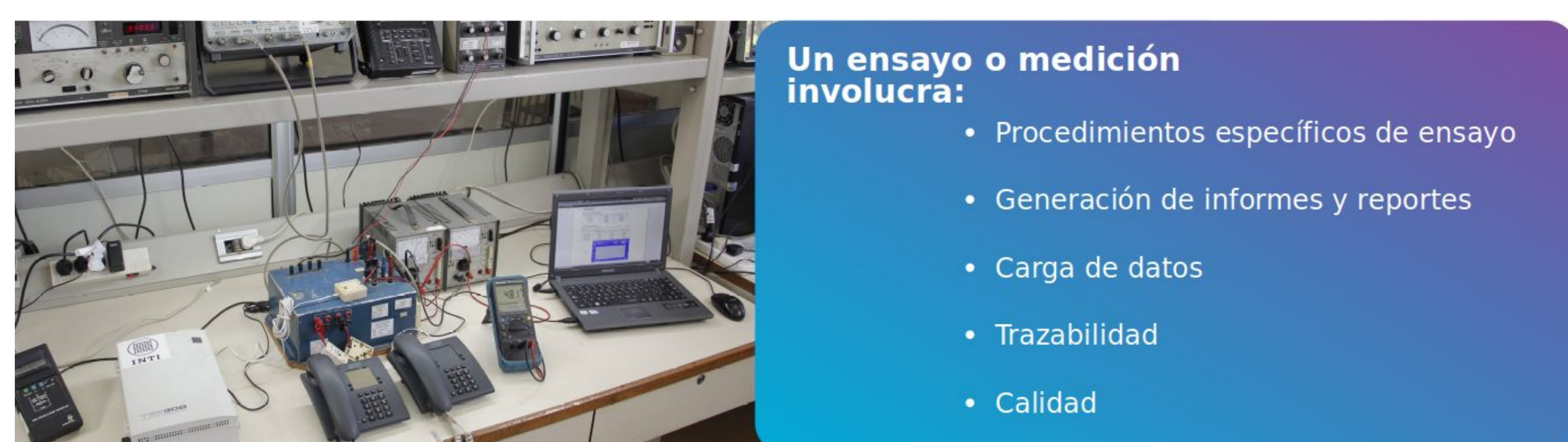
(1)Departamento de Comunicaciones, Subgerencia Operativa de Electrónica y Energía
INTI, Av. Gral. Paz 5445, San Martín, Bs. As., Argentina | ddc@inti.gov.ar

1. Resumen del caso

El Sistema de Automatización de Mediciones (SAM) es una solución modular y extensible que permite la interacción con el ecosistema del laboratorio y el sistema de gestión integral *Tracer*. SAM automatiza tareas repetitivas y sistemáticas, reduce errores humanos y mejora la eficiencia en la generación de informes y trazabilidad de los ensayos.

2. Situación inicial

El laboratorio enfrentaba desafíos con la ejecución manual de mediciones, la generación de informes extensos y la gestión de datos. Estas actividades no solo eran propensas a errores humanos, sino que también demandaban mucho tiempo, limitando la capacidad del personal para enfocarse en tareas más estratégicas y de investigación.

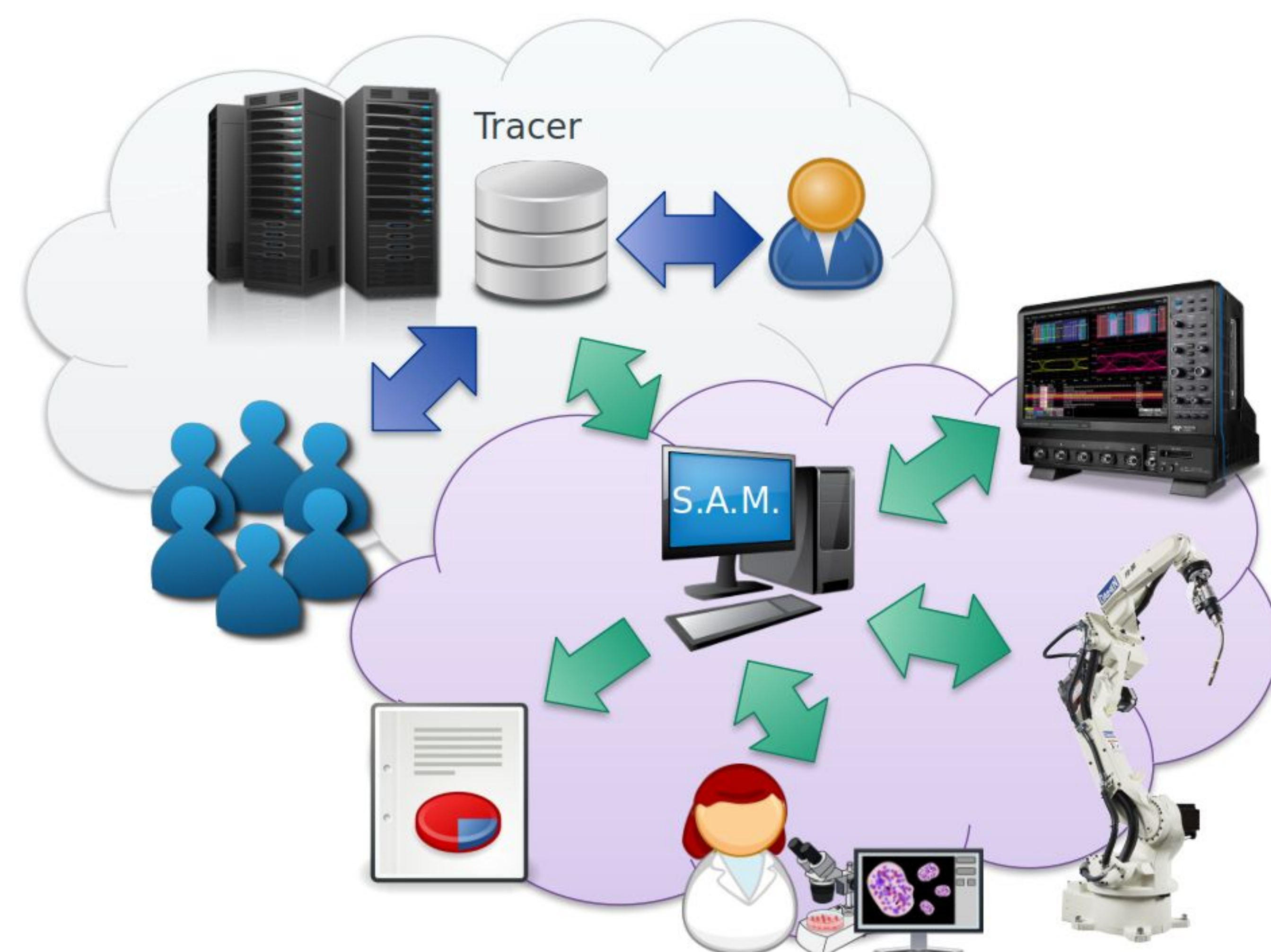
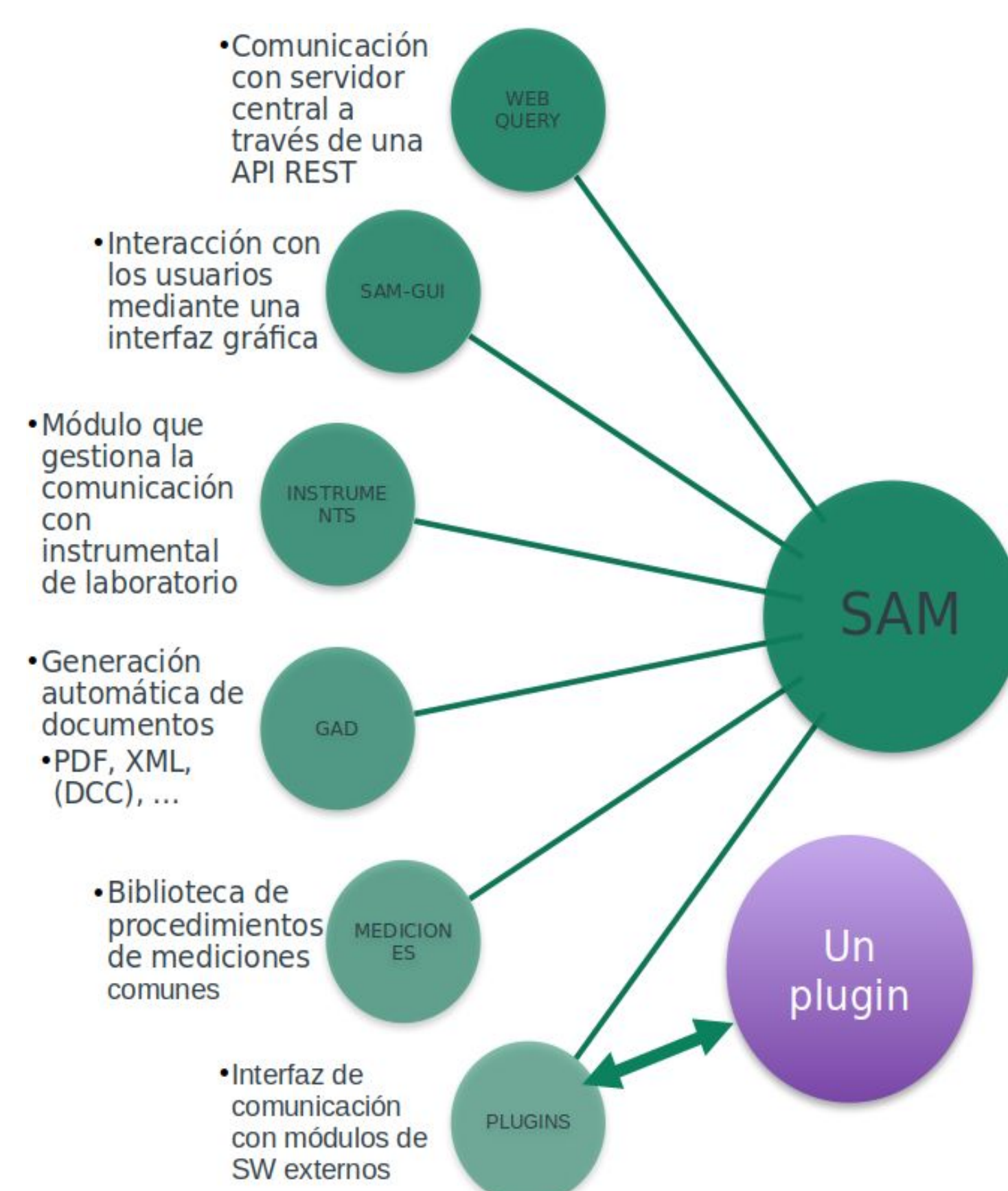


3. Herramientas o métodos utilizados

El SAM está diseñado para ser modular y extensible con *plugins*, permitiendo agregar funcionalidades con mínimos conocimientos de programación.

Los usuarios pueden definir plantillas para la generación de informes y SAM se encargará de crearlos automáticamente, facilitando la documentación de los ensayos en PDF y XML.

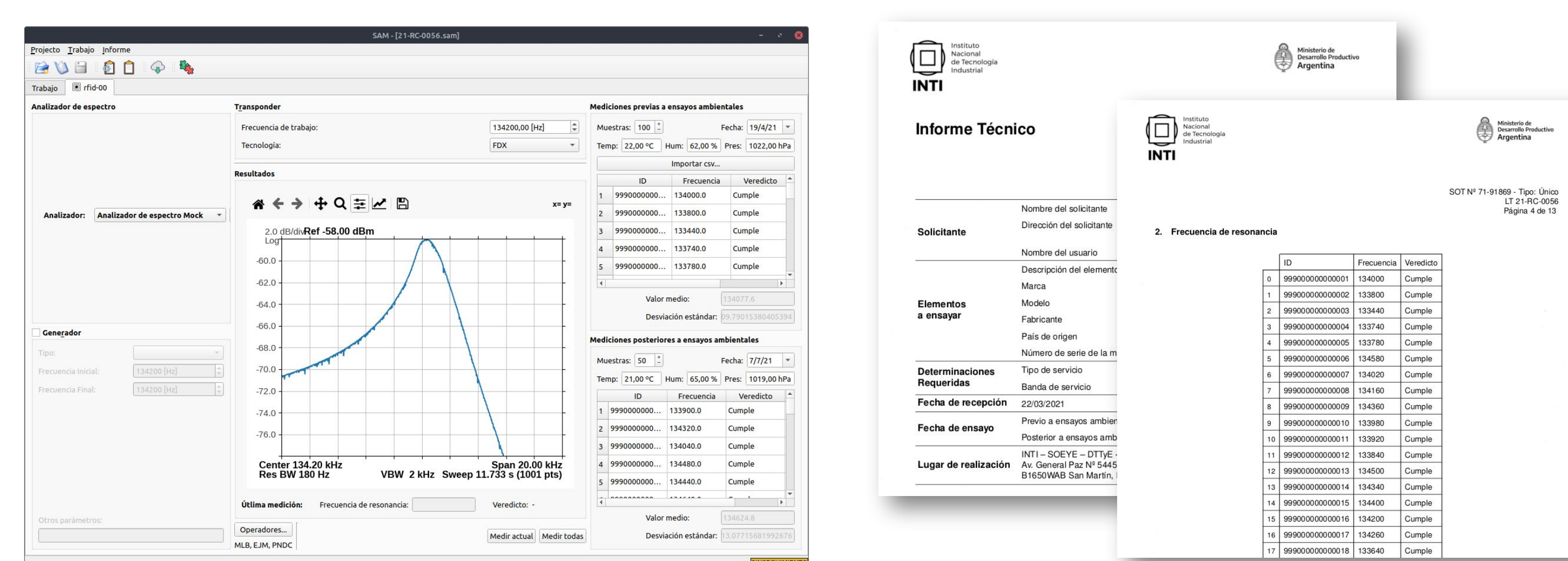
Fue desarrollado con herramientas de código abierto como Python, Qt y PyVisa, asegurando escalabilidad y fácil integración con equipos de laboratorio.



Interacción del SAM con el ecosistema del laboratorio y el sistema de gestión integral Tracer.

Algunas de las herramientas utilizadas incluyen:

- **Python:** flexibilidad y biblioteca extensa para aplicaciones complejas.
- **Qt Framework:** creación de interfaces gráficas y gestión de procesos en segundo plano (*threads*).
- **PyVisa:** comunicación con instrumentos de laboratorio.
- **Latex:** generación de informes automáticos en PDF, garantizando una presentación clara y estandarizada de los resultados.
- **Sistema de control de versiones (GIT):** sistema de control de versiones que facilita el desarrollo colaborativo, con control riguroso de las versiones para mantener la estabilidad ante actualizaciones.



Interfaz de un plugin y los documentos generados.

4. Resultados alcanzados

Reducción de errores: la automatización de tareas rutinarias ha disminuido la intervención del personal técnico, reduciendo significativamente los errores humanos.

Mejor gestión del tiempo: la implementación de la automatización permite que el personal enfoque más tiempo en capacitación y proyectos de investigación y desarrollo.

Facilitación de la automatización: gracias a la arquitectura de *plugins*, el personal puede automatizar tareas diarias sin conocimientos avanzados de programación, al contar con las funcionalidades principales ya integradas en el sistema.

Versatilidad: la estructura modular permite extender el SW para satisfacer múltiples requerimientos, desde la generación de informes o DCCs hasta la comunicación con diversos sistemas o equipos. La arquitectura de *plugins* posibilita la programación de módulos sin necesidad de acceder al código fuente del sistema SAM.

Actualización: la modularidad del sistema facilita la distribución de actualizaciones para paquetes individuales y la adición de características a los módulos sin alterar el núcleo de SAM.

Mantenimiento: cada funcionalidad está claramente definida en los submódulos, lo que simplifica la identificación, solución y publicación de parches.