Matti Vuori, 2013-12-16

RATA project report

Elements of robot assisted test systems

Table of contents:

1. General

Robot assisted test system are not familiar to most people. That is why we take here a look into what components the systems are built of. Understanding the structure of the systems is a key to building – or ordering from a vendor – systems that really meet their goals in everyday testing.

We present the system by its components, because the systems really are systems in the very sense of the word: they consist of many elements, each with a purpose and each of which can be chosen according to any specific needs.

Note, however that the descriptions that follow are generic ones and the practical architectures and designs can vary – and will vary, because the era of these systems is just beginning.

Note also that the terminology for the systems and components varies.

2. Overall view to the system – the elements

The main elements of the test system are shown in Figure 1.

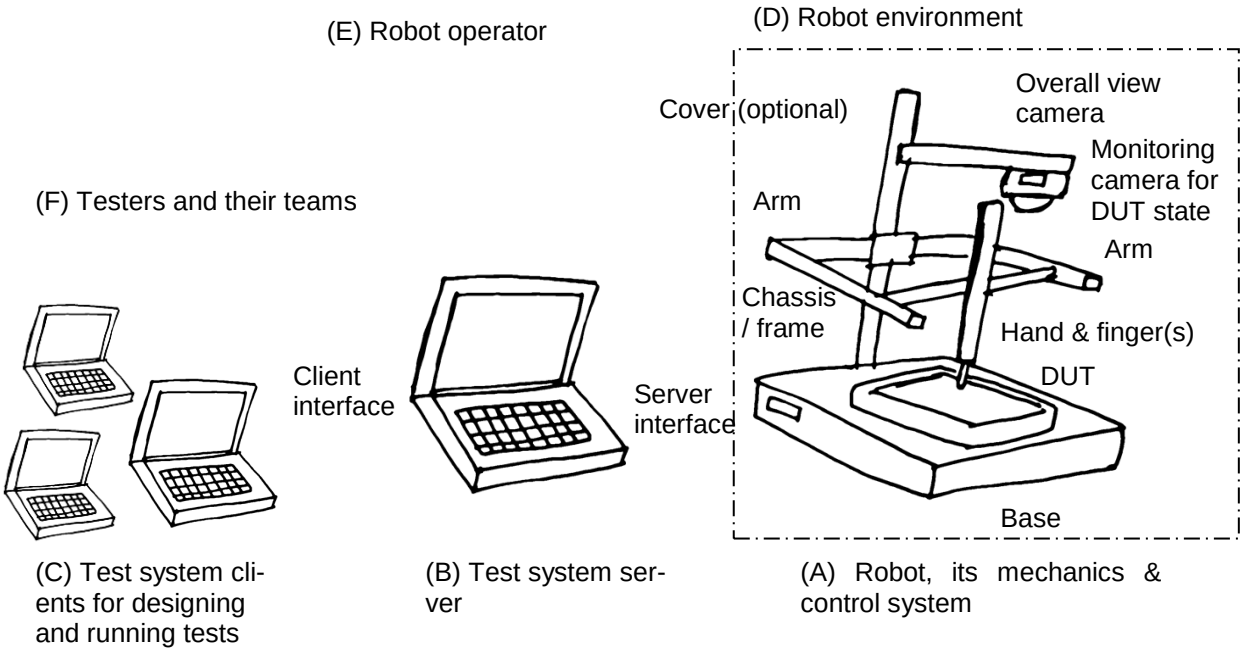

Figure 1. Main elements of a robot-assisted test system.

The main elements are these:

(A) Robot, its mechanics & control system. This is the element that mimics a human by actuating the device / system under test and has a camera that sees from the device display what is happening. It may also have another camera for remote use that shows what the robot system is doing.

(B) Test system server. This controls the robot via the robot's often proprietary low level interface and connects to the camera and other accessories. It provides an API or several to control the robot in an abstraction level that is logical for controlling the device under test. The test clients connect to this computer and use those APIs.

(C) Test system clients for designing and running tests. They contain the test execution tools and usually also the test design tools, which use the test system server as an adapter to the system under test. This computer can be attached locally to the server or be used remotely through the Internet. Why are the server and the client separate? Robots are expensive and there may only be one robot requiring one server – but it may potentially have many users.

(D) The robot system is located in a controlled environment.

(E) The robot system needs an operator to take care of it.

(F) Of course, the testers who design and execute tests are part of this human-technology system.

3. There are variations – for good reasons

There are variations in what is included in the systems and how. And that is good, because:

- Different test system concepts require different features (see paper “Robot assisted test system concepts and their main characteristics”)
- As always, low-end systems are simpler than advanced ones.
- Remote systems need much more system elements than local systems.
- Small companies and especially startups benefit from simplicity, whereas large companies with complex processes tend to prefer more complex system integrated with other complex systems (for the better and the worse…).

These kind are similar to many other “machines”. Consider the simple tools in industry that were later replaced by robot systems. Simple personal systems are by nature as simple as possible offering the minimum amount of technology to get things done. But advanced systems for organisational use are systems that share characteristics from tools, instruments, workstations, manufacturing systems, industrial automation, logistic systems and information systems! That’s a lot of influences and room for both creative design and design problems.

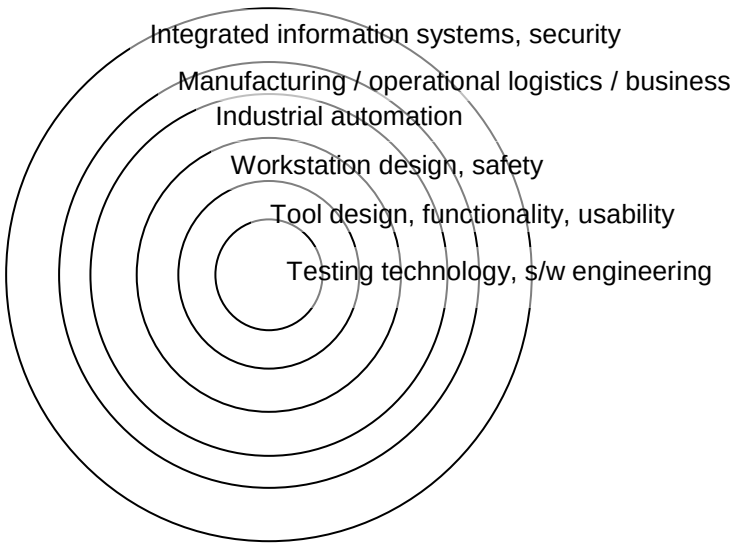

Figure 2. Onion model of the design paradigms and influences.

The good thing is that these systems can really be systems, where all elements can be chosen and changed according to present needs and needs of the future – and different systems can be selected for different purposes.

4. What are included – or can be included – in the main elements?

4.1 Robot, its mechanics & control system

Component	Description and purpose	Variations
Chassis / frame	A structure that holds all the elements in place.	May be a very visible structure, like a structure of a house, or might be reduced to a pillar that holds the robot arm. So, this one varies a lot.
Cover	The system may include a cover for various reasons: • Safety: to block entry to the hazardous area during testing. • Noise control. • Light insulation – to block ambient light entering the camera.	• Fully opaque cover. • Semi-translucent cover. • Fully transparent. • No cover.
Base	Holds the DUT in place. The holding of the DUT needs not be accurate as the system is calibrated before testing, so the server always sees a “straight” coordinate system.	The base or the system of base and robot arm can be fixed or rotate by any axis for testing for example how the device reacts to orientation changes. But those systems tend to be quite expensive
Robot arm	This is like a human arm and just moves a hand or a “finger” from place to place for tapping or other ways interacting with the device. It can guide the finger in rectangular movements, but also in arcs to mimic real hand gestures.	The mechanical construction of the arm can vary a lot.
Finger(s)	Taps of swipes or by other ways actuates the device. Usually needs to be able to limit, control or measure the pressure used.	Materials can vary, because some devices can only detect a finger with given electrical properties. Also, the complexity varies based on the needs. Robots often have only one hour, but to test for example two-finger gestures, two fingers are needed.

Component	Description and purpose	Variations
Hand	When just fingers are not sufficient, a robot may have a whole hand, if the DUT uses gestures that detect hand postures.	This is really something new and a research issue.
Monitoring camera for DUT state; sometimes called test analysis camera	Looks into the display of the DUT (or also for example the keyboard if there is one). Captures video or still images. Usually, still images are used in testing as they can be processed either on the test server or client to find out the state the device is in. It will have a separate, standardized interface to the test server.	The resolution can vary and also other characteristics – and thus the price. The camera can usually not produce a good enough image for testing, but needs to be geometrically corrected and processed. The DUT may need to be isolated from ambient light to reduce glare.
Overall view camera	In remote testing there must be a view to what the robot arm and the whole system is doing. This camera does that. It usually provides a video feed. It will have a separate, standardized interface to the test server. Note that there may still be a need for yet another camera that shoots the whole robot system and its surroundings. That would be more for safety and security purposes.	The image quality is not important. Positioning of the camera may vary.
Robot control interface	An interface to the server. Provides low level access to the robot's hardware.	It may be implemented with a) a proprietary hardware and software system, b) standard hardware but proprietary software protocol or by any other means. A controller card may be needed on the server.
Safety switch	High-speed industrial grade robots require a safety switch to stop them at time of problems or potential accidents.	Low-power, lightweight robots may not need this.

4.2 Test system server

Component	Description and purpose	Variations
Testing API	API for clients, for testing. Optimally abstracts the robot completely and offers the possibility to control the DUT as a human would control a device, that is, not by low-level geometric instructions, but for example by gestures and actuating any UI elements by their name.	• Simple HTTP calls. • Object oriented model of robot and DUT. • Activity based, offering gestures and abstract UI actions
Utility tools and libraries	Specialized tools and libraries for various tasks.	• Image enhancement. • Icon recognition. • OCR.
Robot configuration and control system	Tools for configuring the robot set-up and making this information available through the APIs.	
Calibrating system	Functions for accurately mapping the device UI plane to the coordinate system the test system uses. Preferably as automatic as possible.	• Manual calibrations (match points). • Fully automatic based on analysis of the camera image.
Robot operator user interface	User interface for general handling of the robot.	A GUI with often command line tools (due to needs for low-level system operations).
DUT data management system	Tools to manage any data associated with any particular DUT. Data may be pre-acquired or dynamic.	• Simple stored files. • Linked to configuration management and product data management.
Remote session management system	For remote session, controls the current session and access to the test system.	Sessions can be managed in this server, or by another system – or not at all.
Logging system for testing	Collects logs of everything that happens in the test system. Not to be confused with a logging system in the test client.	As with any logging system, what is logged varies.
Logging system for the robot system	Higher level logging to collect robot system usage data for usage and reliability statistics and to support preventive maintenance.	Varies. The needs depends heavily on the type of the system.
Time measurement system	Collects for example timing and speed related measurements from the DUT. Use of those is optional.	• Simple logging of timestamps associated with actions. • Fully tailorable measurement of selected actions.

Component	Description and purpose	Variations
Support for other measurement system	Interface for integrating other measurements (such as DUT power consumption or any internal measurements) to the test system so that the observations can be associated with testing activity.	• Could be nonexisting, ”logically integrated” with time information or tightly integrated to the test system.

4.3 Test system client for designing and running tests

Component	Description and purpose	Variations
Robot connection tools	Tools to connect to the server, to open a test session and to generally monitor the test server and robot state.	• Scripts to log in. • Browser-based connectivity tools. • Integrated session management tools, with test system reservation, configuration and other tools.
Robot control tools	Tools to control the robot, including a software safety switch and alarm generation tools. Possible actions are limited: a subset of what is possible locally.	• Graphical control UI. • Command-line tools.
Test execution tools	Tools to execute tests on the robot (server) and evaluate their results.	• Simple script executors for written scripts. • Executors for recorded scripts (field recordings or local recordings). • Exploration tools. • Model-based testing tools (usually integrated with test design tools), such as fMBT, TEMA, OSMO Tester.
Utility tools and libraries	Specialized tools and libraries for various tasks.	• Image enhancement. • Icon recognition. • OCR.
Special test evaluation and result analysis tools	Tools for analysing logs, doing root cause analysis and similar. Tools that go further than just check test results.	• Log analysers. • Root cause analysis tools. • Etc…

Component	Description and purpose	Variations
Test design tools	Tools to design and debug tests. Usually integrated with test execution tools.	• Script editors. • MBT modelling / test design tools such as fMBT, TEMA, OSMO Tester. • Imaging tools. • Recording tools.
Test management system	Tools for managing execution of test suites, test configurations, collection and sharing of test results. Needed for complex and repeating testing processes integrated with development or quality assurance.	• From non-existing to full-blown test management integrated with development / QA information systems.

4.4 Other computing resources

Component	Description and purpose	Variations
Other utility servers or cloud resources	We could have some of the work assigned to external servers.	• How about high-speed and high quality image recognition or OCR servers? • Shared icon libraries or font training files in the cloud.

4.5 Test client communications interface

Component	Description and purpose	Variations
Security system	Connectivity between the remote and local systems need to be secure.	• HTTPS. • Virtual Private Network. • Hardware-based systems, including dongles.
Communications protocol stack	Full protocol stack from low level to as high an abstraction level as is needed.	
Session management system	System for managing and monitoring the test system's remote usage. Includes logging of the network activity.	• Non-existent to complex, as for any other type of business or security critical system.

4.6 Robot environment / premises

Component	Description and purpose	Variations
Robot lab / dedicated space	Space for the robot where it can work safely and without interruptions. Access control is required for security (confidential DUT and other client information)	• Space in a development team's room. • Test laboratory. • Remote robot "farm".
Access control system	A managed access control system to the robot premises.	• Local rules for access. • Electronic doors. • Security camera.

4.7 Humans

Component	Description and purpose	Variations
Operator	Local operator responsible for the robot and the server. Carries out robot configuration, DUT installation, monitoring of the system, helps local and remote users with problems. Always accessible by phone. Note: A sign of immature automation is when the operator is needed for trivial helper tasks, such as keeping the DUT alive or resetting the robot if remote users cannot do that.	• May have a workspace in the robot premises or close by. • Dedicated operator or a member of a team.
Testers	People, who develop tests, execute and evaluate them within either development or quality assurance processes. They have specific goals for robot testing and carry out other types of testing, unless they are working in a test laboratory that only does robot testing, but that is rare.	• Testers in development. • Testers in quality assurance. • Developers. • Co-located or remote.
Others	Other people and roles not in the centre of things, but may have a say in the design and service arrangements such as ICT people who arrange access rights, security "officers", managers, consultants, system developers etc.	