

The living knowledge base of your software inner workings. **Automated.**

The Challenge

Today's custom-built applications entail multiple languages, frameworks, and databases. Their growing complexity makes it hard to make even the smallest of changes as developers spend 58% of their time just trying to comprehend the existing code structures.

CAST Imaging

Automatically 'understands' how software work. It visually maps the tens of thousands of code elements and all their relationships into a living knowledge base of the software internals, helping architects and developers navigate the application and quickly find what they need.

Key benefits

Software-intensive companies and system integrators use CAST Imaging to preserve key knowledge, accelerate newcomer onboarding, speed up ongoing maintenance, de-risk modernization and cloud refactoring.

kyndryl.

I can see in minutes what took three months to find before

David Ruggiero
Modernization & Cloud Advisory Leader



15% greater speed and efficiency of development and maintenance

25% faster modernization and optimization for cloud

35% faster ramp-up of newcomers

✓ Faster ramp-up of newcomers

New teammates can quickly learn the application make up, navigate the internal structures, understand the e2e transactions, and literally see all dependencies, without relying on experts or figuring it out on their own. Becoming fully productive much faster.

✓ Safer and faster modernization

Architects can quickly delve into legacy applications, understand existing and to-be architectures, visualize data access and API call graphs, find objects affected by database or framework replacement and candidates for decoupling or microservices. Avoiding dead-ends and wrong turns.

✓ Less time spent making changes

Developers can quickly determine the impact and effort required to make changes or add new functionality, without extraneous consultation or research. They can easily share with others working from anywhere, raising the speed and efficiency of delivering changes.

✓ Automatic knowledge retention

As experts come and go, teams can retain detailed, up-to-date system information for documentation, compliance, and emergency purposes. They can even tag objects and link documents to easily keep track of the logic behind major changes, preserving critical technical knowledge.

Case study

Greater responsiveness and efficiency



Challenge

LTI was transitioning two extensive projects with about a hundred applications to take over maintenance and support. Development teams were spread out and could not work all together at the same place sharing knowledge.

Solution

LTI armed all teams on the projects with CAST Imaging, which automatically created a living knowledge base of the software internals. The architects and developers were able to discover, navigate, and understand all the mechanics of the applications on their own.

Results

LTI measured **20 to 30%** reduction in knowledge transition time for those using CAST Imaging. That was followed by a decrease of rework and coding effort by **30 to 40%**. In addition, due to precise identification of impact of changes, defect density dropped by **20 to 30%**.

Key capabilities

Understands any mix of 150+ technologies. Provides knowledge persistence, exports, REST APIs.

Auto-discovery

- All technologies used
- All data objects and structures
- All code objects and properties
- All frameworks used
- All explicit and implicit dependencies between data, code, frameworks, across all technologies

Navigation

- Zooming in and out, down to the line of code
- Five layers of abstraction
- Logical groupings
- End-to-end transactions
- Intra-application dependencies
- App-to-app dependencies
- Fine-grain, single search across the entire application
- Explanation of algorithms inside objects via ChatGPT



Analysis

- Data access and API call graphs
- Data flows between objects
- Calls between objects
- Indirect paths between objects
- Impact of changes
- Framework dependencies
- Database dependencies
- Candidates for decoupling
- Candidates for microservices
- Architecture design adherence
- Cloud optimization blockers
- OSS risks and structural flaws

Collaboration

- Tag objects against business function
- Annotate and group objects
- Place documents in context
- Create, annotate and share custom views and functional maps

Take a tour castsoftware.com/imaging-trial
Request a demo castsoftware.com/imaging-demo