Engineering the Irish Software Tiger

Kevin Ryan, Lero–The Irish Software Engineering Research Centre

Information and communication technologies, particularly software, play a crucial role in the Republic of Ireland's remarkable economic growth. Successful globalization has posed many challenges and fostered a major strategic investment in research. Ireland's unique position has a major influence on the realization of the software engineering research agenda.

As the "Economic Growth" sidebar describes, Ireland's recent economic growth has been as dramatic as it was unexpected. Yet some might not fully realize the extent to which information and communication technologies (ICT), particularly software, have been central to this growth.

During the past three decades, clusters have been formed in niche areas such as integrated circuit manufacturing (Intel, Analog Devices), telecommunications infrastructure (Motorola, Tellabs, Ericsson), and software localization (Microsoft, Borland, Lotus, Oracle). There are now more than 900 software companies in Ireland directly employing 32,000 people (www.idaireland.com). With total revenues of €12 billion, these firms account for 10 percent of Ireland's gross domestic product.

The Industrial Development Authority states that nine of the world's top 10 software companies have significant operations in Ireland, making it the largest exporter of software in the world. In 2007, Ireland exported €50 billion worth of ICT products and services. Forty percent of the packaged software and 60 percent of the business software sold in Europe originates from Ireland. Subsidiaries of the world's leaders, mainly from the US, dominate this sector, but some indigenous Irish companies have made an international impact, among them Iona, Aldiscon, Havok, and Cape Clear.

Ireland now faces the key challenge of sustaining this remarkable development in a changing world.

NEW CHALLENGES

Where Ireland led, others quickly followed. Ambitious countries and regions in Europe and the Far East all swiftly adopted low tax rates, a focus on improved education and infrastructure, and aggressive development agencies. Because of its success, Ireland is no longer a low-cost country. With a minimum wage exceeding €8 per hour and professional salaries comparable to the richer European Union countries, it cannot compete on price even with many developed countries.

The first effects of Ireland's maturing labor market were felt in manufacturing, with some companies relocating their assembly plants to lower-cost locations in Eastern Europe or Asia. However, relatively low-skill software positions also became vulnerable. The global trend toward outsourcing and offshoring posed a specific threat to the Irish ICT services sector, especially the development, testing, and maintenance of software. Ireland is one of the most open economies in the world, so protectionism would have been pointless, even if the EU allowed it.

Recognizing the changed circumstances and the crucial role that technology would play in Ireland's future as a knowledge society, the Irish government commissioned a national technology foresight exercise in 1998. It set forth the explicit goal of ensuring that "the Government's strategic investments in research, science and technology must be used effectively to underpin Ireland's economic competitiveness and development as a knowledge society" (www.forfas.ie/icsti/statements/tforesight/overview/tforeire.htm).

Economic Growth

The Republic of Ireland gained independence in 1923, and, for the first four decades of its existence, made scant progress economically. Despite being spared the ravages of World War II, it did not benefit from the postwar boom in Europe but continued to be a closed, mainly agricultural, economy until the 1960s. Between 1850 and 1950, one in every two people born in Ireland emigrated. The change since then has been as dramatic as it was unexpected.

Explanations for this success are varied and contradictory, but most commentators would agree that the causes included major strategic decisions by the Irish government, changes in Irish society, and some fortuitous circumstances over which we had no control. Among the major decisions were

- reassessing the economic role of education to view expenditure on education as an investment rather than as an expense;
- opening the Irish economy to international trade by deciding to join the European Union (EU);
- establishing an independent agency, the Industrial Development Authority (IDA), to promote Ireland as a location for foreign direct investment;
- maintaining a low corporate tax rate; and
- investing more than £1 billion in the Irish telecommunications infrastructure so that it was among the most digitized in Europe by the early 1990s.

Circumstances also played a major role.

- English, the first language of almost everyone in Ireland, is ICT's universal language.
- A sudden drop in the birth rate combined with a history of emigration so that, once economic growth accelerated, Ireland's dependency ratio fell from 2.20 in 1993 to 1.24 in 2001.
- Ireland is closely linked by culture, migration, geography, and history to both the US and the UK, two of the major players in the ICT arena.

As Figure A shows, the combined impact of these factors, during a prolonged period of global economic growth, drove a boom in ICT industries and increased Irish economic growth to unprecedented levels.
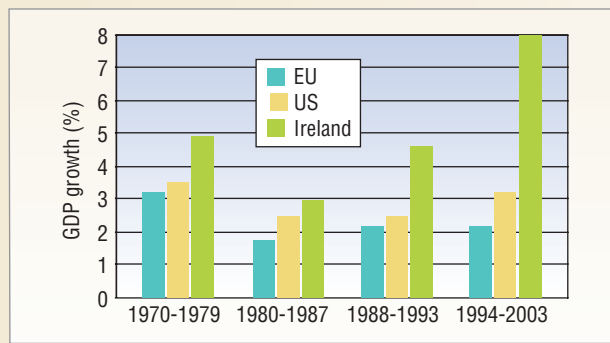

Figure A. Between 1970 and 2003, a combination of factors resulted in unprecedented growth in the Irish economy.

Based on this advice, the Irish government chose ICT and biotechnology as the first scientific areas to emphasize. To reach and then exceed international norms, the government made a long-term commitment to increasing Ireland's research spending from both public and private sources. There was a lot of ground to make up as, during the preceding decade, Ireland's relative expenditure on R&D had hovered around 66 percent of the EU average and 50 percent of the Organization for Economic Cooperation and Development average.

Established in 1999, Science Foundation Ireland (www.sfi.ie) began awarding funds in 2001. One of SFI's first targets was to attract world leaders in the targeted technology areas to move to Ireland. Significantly, David L. Parnas secured a five-year award in 2001 and joined the University of Limerick in 2002 to establish the Software Quality Research Laboratory.

In 2002, following a strategic decision to foster larger collaborative groups, SFI sought proposals for establishing Centres for Science, Engineering and Technology (CSETs). Although loosely based on the criteria for the NSF's National Science Centers in the US, the evaluation criteria for these centers reflected Irish national priorities. Each proposal would involve collaboration between multiple research institutes and industry in specific research niche areas. The objectives were to maximize the funding's impact by forming a critical mass of researchers while at the same time ensuring the research results were of potential value to industry.

In the ICT arena, the first two CSET awards went to the Digital Enterprise Research Institute (www.deri.ie) and the Centre for Telecommunications Value-Chain Research (www.ctvr.ie). SFI also supported smaller groups ("clusters") as well as individuals, whether they were already based in Ireland or planning to move there. To support all of these initiatives, SFI's budget was substantial, even in comparison with larger countries. As Figure 1 shows, SFI continued to commit increasing funding over the first six years of its operation.

Perhaps the most important point about this radical turnaround is that it enjoys wide support in Irish
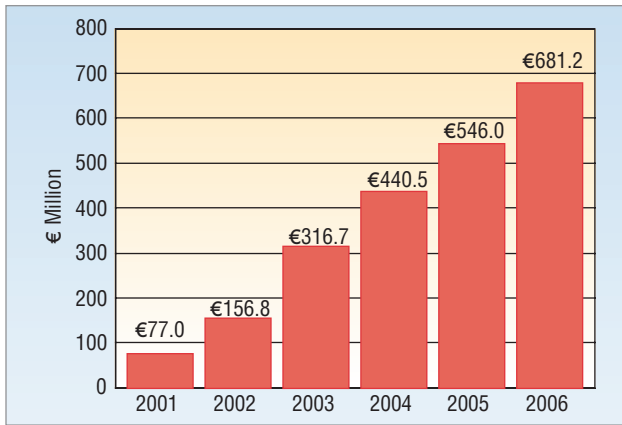
Figure 1. SFI annual cumulative total commitment to research program funding.

society and has been endorsed by most political parties, employers groups, trade unions, and economic commentators. The state-owned Industrial Development Authority (www.ida.ie) has revised its mandate to put a strong emphasis on attracting research-intensive companies to locate in Ireland and to assist existing operations in extending their corporate mandates into R&D. Software is a key sector for this effort.

SOFTWARE ENGINEERING RESEARCH

To thrive in the global marketplace, the Irish software industry must rely on a local research base that is itself internationally competitive. With the support of Science Foundation Ireland, we have set up Lero, a national software engineering research center. The "Lero—The Irish Software Engineering Research Centre" sidebar describes the origins of the center's name. Founded on several specific software engineering research principles, Lero initially involves researchers from four universities: the University of Limerick, Dublin City University, Trinity College Dublin, and University College Dublin.

Software engineering is now a strategic activity in many of the world's largest economic and social sectors, yet software costs, delivery, and quality are notoriously unpredictable. This is happening at a time when industrial and service sectors are increasingly relying on

software for their sustainable competitive advantage and when society at large is increasingly aware of both software's importance and its frequent unreliability. To be worthy of the term, future software engineering will need to demonstrate at least the following characteristics:

- A greater emphasis on well-founded tradeoffs, particularly cost/benefit analysis. This will require better modeling of routine design and close collaboration with industry to ensure continued relevance.
- Equal emphasis on process as well as product. Both must be subjected to thorough analysis, rigorous modeling, and experimental validation.
- An effort to deliver both flexibility and reliability over a software artifact's lifetime. Historically, when software has been shown to be highly reliable, we have been understandably reluctant to change it. Where software was constantly modified, we accepted a reduced level of reliability. In the future, rapidly evolving systems will require both flexibility and reliability.
- Greater specialization by application domain. Increasingly, developers recognized that generic tools, techniques, and models must be supplemented by those that embody the rules, constraints, and particular details of the environment in which the eventual system will be deployed.

Within the context of a small country seeking to maximize its national impact, software engineering research should therefore be based on four key principles:

- Take a comprehensive approach to software engineering.
- Focus on a few key application domains.
- Plan to trial innovations, whether of process or product, in a realistic industrial setting.
- Develop relationships that promote and reward collaboration.

LERO'S RESEARCH AREAS

On this basis, Lero has developed research in five interrelated areas, each of which contributes to the overall objective of making software that is both flexible and reliable. Lero's initial focus is on automotive systems, medical devices, and financial/enterprise systems—domains that have great potential for Ireland. The automotive sector is of special interest as it is one of the most dynamic and rapidly growing in the world. A recent study by Mercer Consulting estimated that the world market for automotive software would be worth more than €130 billion by 2015.

Global software development

Global software development (GSD), the practice of distributing software development activities over geo-

Lero–The Irish Software Engineering Research Centre

Lero is a brand name, not an acronym. Establishing a strong brand, which would represent the collective identity, was deemed essential for a multiuniversity venture. Lero was selected with the help of a professional branding agency. The name was suggested by an old Gaelic idiom, léaró dóchais, which means, appropriately enough, "a ray of hope."

Table 1. Global software development—benefits (+) and challenges (−).

Process	Temporal distance	Geographical distance	Sociocultural distance
Communication	+ Time zone effectiveness – Delayed communication – Delayed feedback	+ Proximity to market/customer – Lack of informal communication – Dependency on ICT – Increased effort to initiate contact – Providing technical infrastructure – Cost of travel	+ Innovation and shared best practices * Asynchronous communication preferred by non-native speakers – Language differences and misunderstandings – Managing frames of reference
Coordination	* Time zone efficiency – Reduced hours of collaboration – Synchronized team meetings difficult – Availability of technical infrastructure – Coordination complexity – *Modularization of work* – *Lack of mechanisms for creating shared understanding* – *Management of project artifacts*	+ Access to large labor pool * Standardization in work practices + Allocation of roles and team structure – Reduced trust – Lack of awareness/team spirit – Modularization of work – Lack of mechanisms for creating shared understanding – *Coordination complexity*	+ Mix of skills and experiences * Language and cultural training – Lack of domain knowledge – Doubtful of others' capabilities – *Lack of mechanisms for creating shared understanding* – *Standardization in work practices* – *Coordination complexity* – *Lack of awareness/team spirit*
Control	– Management of project artifacts + *Time zone effectiveness*	– Lack of concurrent engineering principles + *Allocation of roles and team structure*	– Perceived threat from low-cost alternatives – Adapting to local formalized norm structures – Different perceptions of authority/ hierarchy

* Both a benefit and a challenge; italic type—common factors across dimensions

graphically separate sites, whether they are part of the same organization or not, can exist in many forms. A particular phase, such as testing, can be conducted at a remote location by a subsidiary of the same organization or by a third party. GSD typically involves established companies owning or working with development centers in India, China, Eastern Europe, and other locations.

While GSD's potential benefits are easily stated—for example, lower cost, "follow the sun" development, easier localization, access to highly skilled labor—the realization of these benefits has proven difficult in practice. As Table 1 shows, GSD also raises problems of coordination, communication, and control of the overall process due to the three "distances' involved: geographical, temporal, and sociocultural.[1]

Research in GSD has focused on some particular tasks, chosen partly because of their special relevance to Ireland:

- Study the relationship between agility and GSD. Intuitively, there is a direct contradiction between the two, but tailoring and enhanced communication can deliver many of the benefits of agile approaches within virtual teams.
- Analyze the reasons why open source software, an example of successful GSD, works and gain insights

that might be applicable both inside and outside corporate structures.
- Consider how small and medium-sized enterprises (SMEs) can benefit from GSD by adapting some of the processes that have worked for larger corporations, including participating in global OSS development.
- Determine which practices and approaches work well for some particular domains or enterprise types.
- Study the social, organizational, and cultural factors that influence the success of distributed, multicultural development teams. Conceptualize and analyze the emergent communication patterns and difficulties to reduce the risk of failure.

In the Irish context, "two-stage offshoring" has emerged as a development model for GSD. Working within a multinational corporation, Irish development sites act as a bridge between headquarters and a low-cost location. Ireland is both a source and destination for global offshoring, making it an ideal place to conduct this research.[2]

Autonomic systems

The Autonomic Computing Initiative aims to derive inspiration from the human autonomic system—and, on a broader scale, from many other aspects of

Industry-University Linkage

Established as NIHE Limerick in 1973, the University of Limerick devised one of the first electronics degrees in Ireland. Analog Devices Ireland (ADI), a leading integrated-circuit provider, was attracted to Limerick in 1977, partly because of the electronics focus at UL. The relationship between the organizations was fruitful for both. UL graduates and, later, postgraduates, found placements and employment at ADI, which in turn supported UL with equipment, guest lectures, and research funding. Because ADI is one of the few vertically integrated foreign direct investment operations in Ireland, the research relationship was essential to its survival. The symbiosis is illustrated by the fact that the first head of UL's electronics department eventually became worldwide head of quality for ADI, while ADI's first research director was later a professor at UL.

biological systems and nature—with the goal of developing self-managing, and ultimately self-governing, systems.

Lero's focus has been to develop tools and techniques to ensure that self-managing software is predictable at runtime. Lero's research has aimed at developing

- model-based approaches to describing adaptive behaviors;
- visual analytic metrics to describe software and sensor data navigation, aggregation, and exploration; and
- novel decentralized collaborative optimization techniques for autonomic components.

We have been examining the role of autonomic systems and autonomic communications—the application of such techniques to achieve self-managing networks, middleware, and so on—in various applications and domains, including our center's focus domain of automotive software.

Software as a service

Realizing the software-as-a-service paradigm will require a flexible deployment model that supports the dynamic and large-scale integration of heterogeneous service offerings. This poses major research challenges including

- composing services within a strict time boundary,
- adapting and combining services from numerous providers, and
- integrating these services into heterogeneous target environments.

In keeping with the focus on domain orientation, the research approach within Lero is to combine two general software engineering technologies—model-driven engineering (MDE) and aspect-oriented programming (AOP)—within the context of emerging automotive communication systems. MDE aims to cope with complexity by using domain-specific languages (DSLs) and to add flexibility by automating transformation steps during development. The transformation process will use AOP techniques in composing the appropriate behavior models to fit each specific target environment.

Mathematics of software engineering

The complexity of software in many domains, including Lero's focus domains, makes it impossible to guarantee quality through testing. However, various mathematical approaches, suitably tailored, can greatly increase confidence levels and ensure quality software.

The Lero approach has been to understand abstract mathematical principles underlying software, adapt them, and integrate them into the entire software development life cycle, increasing our levels of confidence in the software's reliability and correctness.

Current research includes the following:

- Development of a methodology for hybrid systems (exhibiting both discrete and continuous states) that will assist the user in developing a specification that not only captures the embedded software's behavior but also captures a description of the environment's continuous dynamics that is not only being controlled by the software but is also interacting with it.
- The use of software verification techniques and tools to analyze the specifications of various features and to detect when possible interactions might occur. While these features might operate successfully independently, unexpected interactions might occur when they are integrated, particularly into legacy systems on which newer software might be built.
- The different types of dynamic dependencies in service-based system architectures and their orchestration and interaction processes go beyond the current solutions for static and structural connectivity dependencies. Lero has been investigating the use of graph theory to provide a formally sound and effective integration solution for this increasingly popular system development paradigm.

Software product lines

Software product lines (SPLs) are defined as "a set of software-intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed

way" (www.sei.cmu.edu/productlines). Their use has become quite commonplace in the automotive domain, with software platforms being customized and used in several hundred products simultaneously, and in medical devices—two areas in which Lero has interest.

With SPL, we are exploring the potential for visually informed variability management. We believe that systematic variability management and product derivation can, and should, be augmented by visualization techniques and tools that support the understanding, management, and effective use of product-line development artifacts, their built-in variability, and the dependencies among them. With suitable techniques, such visualizations can also amplify the cognition about large and complex datasets created and used in industrial software product line engineering.

Seamless model-driven development with a flexible combination of interactive tools and automation requires semantically rich models. Applying model-driven approaches in an SPL context requires languages to describe typical SPL aspects—variation points, variants, features, configurations, and realization of variability. These modeling languages also must support the different conceptual levels of domain and application engineering. As part of our research, we are developing modeling languages that developers can use to describe the models they process in model-driven approaches. Applying model-driven SPL approaches in the context of a specific domain—for example, automotive engineering—also requires languages to describe domain-specific knowledge. Models described in those modeling languages can be used as a foundation for interactive tools or automation.[3]

COLLABORATION BENEFITS

Collaboration between institutions and research areas doesn't happen by accident and always incurs some overhead. Organizational resistance must be overcome, and people must be willing to move outside their comfort zones. Individual and institutional commitment to a higher ideal are necessary and sufficient to get collaboration started. To continue and prosper, the collaboration must show results for both the participating individuals and their institutions.

Examples of the benefits to Lero's members during its first two years included the following:

- Participation in major international initiatives such as the Networked European Software and Services Initiative (www.nessi-europe.com), leading to greater involvement in EU-funded research projects.
- Success in national funding competitions to establish collaborative graduate education programs in software engineering. This in turn strengthens student recruitment both at home and abroad.

- Increased visibility attracts international conferences. Many of the major conferences in Lero's research areas have been, or soon will be, held in Ireland.

Over the longer term, we can expect to see additional collaborative projects and other initiatives aimed at securing the relationships and maximizing the benefits from being involved in a larger, more visible consortium. For its part, the Irish government remains firmly committed to an ambitious program that includes, for example, a doubling of PhD output by 2013 (www.entemp.ie/science/technology/sciencestrategy.htm).

This account of Ireland's past achievements and current ambitions in the software area is intended to inspire others to do the same. Small countries or remote regions can show that while the world is not flat, they can at least learn to deal with the slopes.

Success depends on many factors, but the following are essential: a united vision, long-term commitment, and a clear focus. There can be no segregation of political, industrial, and academic activities. The "Industry-University Linkage" sidebar illustrates the sort of staff mobility that is possible between industry and universities. Smallness can confer the advantages of agility and flexibility, which are increasingly necessary in a rapidly changing world.

Finally, cooperation among universities, industry, and academia is fundamental and fruitful in helping to maximize the impact of necessarily limited resources. On that basis, Irish software engineering looks forward with confidence to making a major contribution to the country's social and economic development. ■

References

1. H. Holmstrom et al., "Agile Practices Reduce Distance in Global Software Development," *Information Systems Management*, vol. 23, no. 3, 2006, pp. 7-18.
2. H.H. Olsson et al., "Two-Stage Offshoring: An Investigation of the Irish Bridge," *MIS Quarterly*, vol. 32, no. 2, 2008.
3. G. Botterweck, L. O'Brien, and S. Thiel, "Model-Driven Derivation of Product Architectures," *Proc. 22nd IEEE/ACM Int'l Conf. Automated Software Engineering* (ASE 07), ACM Press, 2007, pp. 469-472.

Kevin Ryan is the director of Lero–The Irish Software Engineering Research Centre, a multiuniversity collaborative research center headquartered at the University of Limerick, Ireland. Contact him at kevin.ryan@lero.ie.