The challenge of creating an efficient, integrated Educational Technology architecture in the business of teaching and learning in the 21st century

Alexander Roche MA App Ling, BSc

Abstract

This paper explores an approach to defining and implementing a best-practice Educational Technology systems architecture in support of the business of teaching and leaning for an Educational institution. It examines the challenges institutions face in welding disparate and often highly commoditised software applications into a unified whole capable of effectively serving the requirements of stakeholders across the business of education from student acquisition, to student records management, to teaching/learning facilitation, to compliance/quality and content management.

In an architectural exemplar, the paper looks closely at how a good learning object repository can be used as a content management hub to 'switch-board' content from different sources directly into the core business of the delivering high quality teaching-learning in a increasingly online world.

Table of Contents

1. Objectives of this paper

The core objective of this paper is to explore the challenges faced by an Educational institution in conceiving, designing and implementing a robust and effective Educational Technology Systems Architecture in support of the business of teaching and learning. It will examine the historical context in which this complex problem space has formed and will seek to expose some of the traps and pitfalls that have beset the Education sector as the various challenges have been faced over the last decade and more. It will look at some of the integration technologies available and where in an architecture they might be used. It will also look at an example of the changing architectural landscape as systems have proliferated. Finally, the paper will look to the future and explore the architectural impacts arising from content management developments coupled with improving content federation standards in a modern architectural exemplar combining both intra and inter-institutional opportunities.

2. Why is good systems architecture important?

Where disparate systems are in operation and there is the inevitable requirement for the exchange of information across systems, a coherent, robust and efficient systems architecture becomes compelling. This is no less true for the Education sector and specifically for Educational Technology systems. If this is not present then each system will be less capable of completing its designated information tasks or the Education institution's staff, teachers and /or students will be burdened with manual updates and duplicate data entry and so on leading to additional operational costs, quality assurance and sanity assurance issues.

Unfortunately, good system architecture integration is the exception rather than the norm. It is common to see Educational Technology systems deployed without integration and unnecessary and onerous burdens placed therefore on the staff, teacher and student populations.

It is also very common to see Educational Technology architectures completely lacking in effective content management. With content making up one of the most expensive components of education, this is an area that many Educational institutions could improve upon.

Deployed technology should also be an enabler serving the needs of the Education process. Poor systems integration unvaryingly yields poor usability which is in opposition to the way we would see Educational Technology used. Further, good architecture means that replacing systems and adding new ones is far simpler, more cost effective and basically more feasible.

It is reasonable to argue that the standard of systems integration architecture in the Education industry and, in particular, Education Technology systems integration Architecture is on the whole far below a level where the full potential of deployed systems can be leveraged and where optimal usability for users has been achieved.

3. The emergence of Educational Technologies at a systems level

Education is a complex and an information rich domain and so it is unsurprising that the advent of computers and their dissemination to the levels we see today has led to a proliferation of specialist information systems. These information systems have been used to manage the business of Education both in terms of improving the process of teaching-learning and the processes required to allow the teaching and learning to proceed.

It is relatively recently that these system level technologies have emerged. This recency has been a strong contributor to the challenges faced by Education institutions in creating an efficient systems architecture as the knowledge and expertise required has been only partly available both in the implementing institution and also in the industry as a whole.

Some of the earliest examples of Educational Technology systems include the Learning Management System (LMS) and the Student Management System (SMS). To this we have seen the addition of the Learning Content Management System (LCMS), synchronous learning environments, ePortfolios, Logistics management systems, resource booking systems, etc. There are now many other applications within the space.

Example Educational Technology systems

- Learning Management System (LMS)
- Student Management System (SMS)
- Learning Content Management System (LCMS)
- ePortfolio
- Collaborative environments such as wikis, groupware, social networking applications

- Curriculum Management system
- Library catalogues and collections
- Customer Relationship Management system (CRM)
- Assessment systems
- Resource Management
- Quality Management Systems (QMS)
- Enterprise Resource Planning system (ERP)
- Logistics system
- Anti-plagiarism system
- Bibliography Management system
- Synchronous classroom
- Content Authoring tools

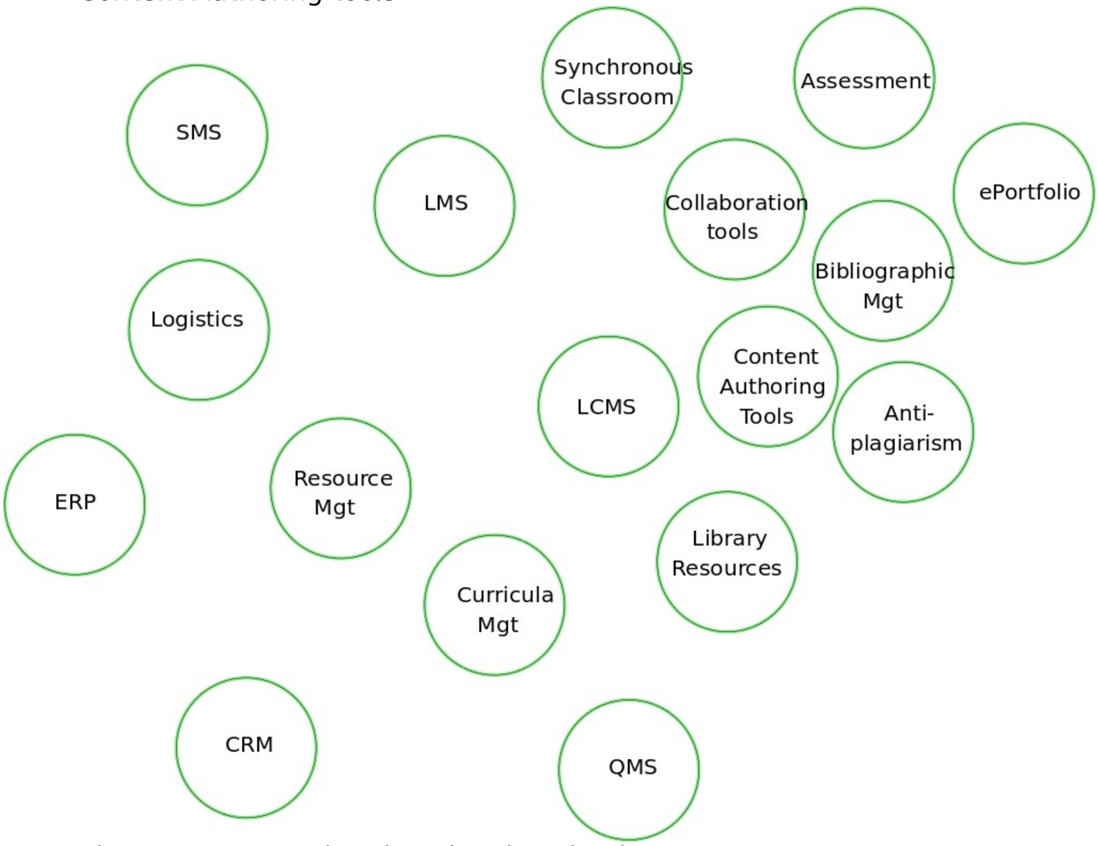

Figure 4.1: Example Educational Technology systems

3.1 The Learning Management System

Perhaps the most broadly known is the LMS (also known as 'Virtual Learning Environment'/'VLE' or 'Course Management Systems'/'CMS'). The LMS is effectively an online system for facilitating the process of teaching-learning including access to resources, assessments, grades, teacher-student and student-community communications and so on.

LMSs are complex, feature rich and highly commoditised in the sense that the feature set is well 'understood' in the market much as a toaster is understood to to be a 'griller of bread' and not, for example, a 'griller and baker of bread'. The

LMS also emerged and reached maturity as a system in the 1990s quite quickly with most being developed by dedicated software vendors rather than being developed in-house by the user institutions themselves. This contributed to the rapid commoditisation as the requirements were distilled across the industry instead of solely from specific institutions alone. Commoditisation has also meant that Open Source developers have been able to subsequently and rapidly enter the market and compete perhaps more effectively than the original proprietary software development companies. This could be because functional innovation is decoupled from the need to manage versions across a client-base and to share financial resources between sales and marketing and product development.

The LMS was originally conceived effectively against a virtual school analogy and was also historically conceived as a stand-alone system. This has meant less support for integration standards and relatively higher levels of difficulty in terms of broader systems integration.

LMSs are not whole-of-business systems. For example, they do not manage content effectively but are rather 'content dumb' . Neither do they manage the complexities of student records management with the detail generally required. They also do not as yet contain many teaching-learning tools at level of sophistication similar to other specialist systems. LMSs therefore need to be deployed in an integrated partnership with other systems that manage other aspects of the business of Education.

Leading examples of LMSs include the Open Source Moodle and Sakai and the proprietary Blackboard and Desire2Learn systems amongst many others. Blackboard remains the most popular in the US. Moodle is one of the most powerful and most usable, is perhaps the most popular on a world-wide basis and continues a meteoric rise in popularity. Each of these examples present strengths and weaknesses.

3.2 The Student Management System

The Student Management System or SMS (also known as a 'Student Information System'/'SIS') manages student records from registration, enrolment, personal information, recognised prior learning, compliance reporting and so on. They usually manage the financial records associated with study and can incorporate aspects of Customer Relationship Management systems to allow lead management and account management and communications.

The SMS emerged earlier than the LMS but has been less visible and indeed probably less commoditised perhaps because of that lower visibility and because they were frequently developed in-house to quite localised requirements. In-house development has meant that the quality of these systems varies widely. This is perhaps because Educational institutions rarely do software development as well as dedicated software engineering companies. Development technologies are often poor in such instances with domestic class technologies such as Microsoft Access and FileMaker Pro being used rather than more robust and scalable technologies.

Lower levels of commoditisation for the SMS also means other systems tend to duplicate functionality. This is certainly the case with the LMS which generally offers a subset of the student record management features offered by a more specialist SMS. This also introduces challenges in a systems integration as duplicate functionality needs careful management.

Similar the LMS, SMSs have generally been conceived as stand alone systems. Typically they have been developed and implemented to solve a particular information management problem, that is student records management, in isolation with the broader teaching-learning needs of an institution. This has meant that the need for a standards-based approach to facilitate inter-operation with other systems has been less compelling and therefore they are harder systems to integrate into a broader Educational Technology architecture.

Integration is generally compelling with the LMS to facilitate enrolment data in one direction and assessment/course completion data in the other. Frequently the SMS is integrated 'beneath' the LMS so that users have a single interface to interact with. In these circumstances the integration is generally much richer with personal information and even financial transaction data being passed between the systems.

Leading examples of SMSs would be RM's Integris (though that product is very K-12 focussed) Calista, Paradigm and so on.

3.3 The Learning Content Management System

A relative latecomer to the Educational Technology space is the Learning Content Management System or LCMS (also know as 'Learning Object Repository' or 'Content Object Repository'). The LCMS manages content in the modern content management sense, that is it can effectively control content version, content state (draft, live, archived, review, suspended, etc). It allows content to be created, refreshed and aggregated and disaggregated in collaboration with others as well as to publish the content for consumption. This is the standard role of the Content Management System or CMS and there are literally hundreds of examples of conventional CMSs available on the market today. The LCMS is different however as it is an Education industry focussed CMS. This means that it should be able to manage content management challenges specific to the Education sector out-of-the-box whereas a generic CMS would require often expensive and time consuming customisation. Examples of the these Education sector content management challenges include the ability to manage:

- discovery and publication into other Educational Technology systems such as LMSs
- digital rights
- Educational Technology standards including content and user data standards such the Sharable Content Object Reference Model and its constituent standards, the METS standards, etc.
- educational design standards such as the course sequencing standards in the SCORM
- specialist Educational content types such as assessments
- federation with other Education sector resources such as libraries and academic journal collections using standards such as OAI and z39.50, etc.

The LCMS arose specifically to fill the gap left by the commoditisation of the LMS without content management. This has meant that LCMSs traditionally facilitate integration with LMSs making their integration into the broader systems architecture easier.

There are still very few mature LCMSs available. There is however on exceptionally good one available, Equella developed by Australian company The Learning Edge. It is very well engineered and effectively defines best practice in this area of an Educational Technology architecture.

3.4 ePortfolio

A system to aggregate content and share and display it in different views for different purposes.

Common integration points include with the LMS as an extension of its teaching-learning capabilities including assessments. Integration with job placement systems for presenting resumes and portfolios of work is also not uncommon.

A leading example ePortfolio system is the Open Source Mahara. Desire2Learn have also developed a contender in this space.

3.5 Collaborative environments such as wikis, groupware, social networking applications

Various often Web 2 applications and technologies varying from collaborative publishing tools such as wikis to blogs and social interaction tool-sets.

Common integration points include the teaching and learning portal for example via the LMS and with content production systems and processes in the LCMS and in content authoring tools.

3.6 Curriculum Management system

A system to describe and manage the structure and description of curricula including meta-content such as learning outcomes, assessment models and so on. These do not generally contain the learning content developed to support the curricula. These systems are still effectively rare.

Integration points include the LCMS for content production mapping and the SMS for enrolment and outcomes mapping.

3.7 Library catalogues and collections

These involve established collections of academic and related materials such as journals, news materials and so on. These are often referred to as 'databases' but are effectively content collections. Traditionally these have been accessible via commercial subscription models around an institutions student and full time employee (FTE) numbers. Leading examples include GALE, Proquest, Informit, EBSCO, etc.

3.8 Customer Relationship Management system (CRM)

CRM systems facilitate the sales and marketing process through close account management of student candidate leads. They frequently include mass marketing and data mining tools designed to increase the effectiveness of the sales and marketing process. They also frequently cross over with SMSs in terms of records management.

An example of an SMS is the Open Source Sugar CRM.

3.9 Assessment systems

Assessment systems manage the process of delivering assessments and their outcomes. They usually include assessment authoring and importation tools. They are specialist tools and generally extend similar capabilities present in LMSs. Deployment with LMSs is generally difficult due to this cross over of functionality but not impossible. Some have clean integrations available for LMSs such as Hot Potatoes within Moodle for example.

3.10 Resource Management

Resource management systems are used to manage usually physical resources such as classroom and teaching resources such as data projectors, etc. Most would include class scheduling capabilities as well.

3.11 Quality Management Systems (QMS)

QMS are basically document management systems designed to control key compliance documentation. These are often implemented on low level technologies and could easily be subsumed by more powerful, configurable content management systems such as LCMSs.

3.12 Enterprise Resource Planning system (ERP)

ERPs allow an organisation to closely control resource purchasing and related expenses. They are generally huge systems with very broad functionality and are frequently implemented with little thought to broader, specialist systems.

3.13 Logistics system

Logistics systems in Education are usually used to control the despatch of hard copy resources such as textbooks and other paper-based materials or possibly CDROMs and similar offline media. Logistics systems are commonly integrated with or even part of the SMS as despatch is usually triggered by some part of the enrolment process.

3.14 Anti-plagiarism system

These systems are effectively databases of academic content that can be programmatically compared to submitted materials, usually in the form of submitted assessment materials or research. These systems are commonly integrated with assessment systems, either specialist assessment systems or assessment engines within LMSs. They are also frequently used manually by markers, that is, not integrated at all.

A leading example is the TurnItIn system.

3.15 Bibliography Management system

These systems allow the management of academic reference materials. They usually contain functionality to collate and format a bibliography against a range of selectable reference standard such as Harvard, etc.

An example of such a system is EndNote.

3.16 Synchronous classroom

If an LMS was conceived against the analogy of a school, these systems were conceived as virtual classroom in which teaching-learning can be conducted as close as possible to that of a face-to-face environment. They usually include inter-user video and audio plus sharing tools so that user can interact with the same content and see each other's input.

Examples of leading systems include Adobe Connect, eLuminate, Interwise, Centra and Webex amongst many others.

3.17 Content Authoring tools

Content authoring tools are many and diverse but are a legitimate Educational Technology system. They can be immensely complex or relatively simple. They include the ability to aggregate content into some format of learning object, generally with publication pathways. Most include export options including formatting as SCORM content aggregation packages including often as SCORM SCO packages with on-board functionality capable of interoperating with a SCORM conformant LMS.

4. Systems commoditisation and the impact on architecture

System commoditisation has an enormous impact on systems integration architecture as it can effectively dictate the integration boundaries. In a commoditised application space, apart for very rare cases where a software vendor can break the commodity paradigm and create some new application that subsumes an old one, they must follow the expectations of the user community and conform to the functional requirements, perhaps enhancing functionality where possible> Alternatively they must must build systems that sit outside those functional boundaries, effectively different systems entirely.

If development is of a different system, outside of a related but highly commoditised application, it is then likely that integration with that application will be compelling for the software developer. The close integration of LCMSs with LMSs is a good example of this. Thus, through this process, the boundary lines of applications as commodities become more clear and more fixed.

Extending the toaster vs bread maker example: in a market where there exist 'bread griller' toasters with high distribution and 'bread baker' bread makers with lower but still reasonable distribution, a manufacturer would be very brave to make a combined 'toaster-bread maker'. This is equally true for Educational Technology systems.

5. The difficulty in conception, design and implementation of a good Educational Technology Systems Architecture

The lack of well designed and developed Educational Technology systems architectures in the Education sector has a composite causality. Firstly there is the historical context of recently emergent technologies such as the Internet and web-based software. Standards and methodologies are not as defined and established as they will be. Bespoke approaches to integration require much more tenacity, technical and business skill and, frankly, luck to successfully implement than an accepted standards-based approach. These issues contribute to greater difficulty in establishing good architecture.

Secondly, several of the key Educational Technology applications were developed independent of broader context. This has made them harder to integrate. This Independent deployment has also seen many organisations take on low quality processes and quality assurance in order to manage the shortfall. Such solutions can become difficult to replace once established.

Thirdly, technology in the Education sector has not been traditionally an area of either perceived or real core competence. Leaders in Education usually come from the academic and/or teaching disciplines and rarely from IT backgrounds. Projects frequently look at particular and relatively localised outcomes often with relatively short return on investment cycles, certainly compared to other industry sectors. Establishing good systems architecture is generally a longer term ROI.

Fourthly, IT resources in the Education sector, as per many sectors, revolve predominantly around desktop and communications infrastructure. Applications expertise, specifically expertise in key Educational Technologies, has not necessarily been part of the growth of IT capability. When Educational leaders need advice on Educational Technologies however, they will generally turn to their IT teams for advice. They may not be in a position to provide it and this may not be evident to the institution's leadership.

Fifthly, the key to a good architecture is excellent selection and integration of systems against well identified business requirements and this, almost without exception, requires significant effort. For example, unfortunately, Educational institutions frequently and seriously underestimate the effort required in an integration project and/or fail to assign the right resources to it as a priority.

Lastly and also unfortunately, there is generally an incomplete awareness of the existence and purpose of the various Educational Technology systems amongst Education industry practitioners including their leaders and there advisors. Many practitioners in the Education domain are unfamiliar with the existence of the broader set of systems which has meant that historically systems have been implemented in a piecemeal fashion and without broader systems architectural considerations being addressed.

6. An example of the historical evolution of Educational Technology Systems Architecture

A description of the historical evolution of Educational Technology systems integration can be described as follows:

1. Stand-alone systems, relatively few Educational Technology systems deployed: effectively no integration.

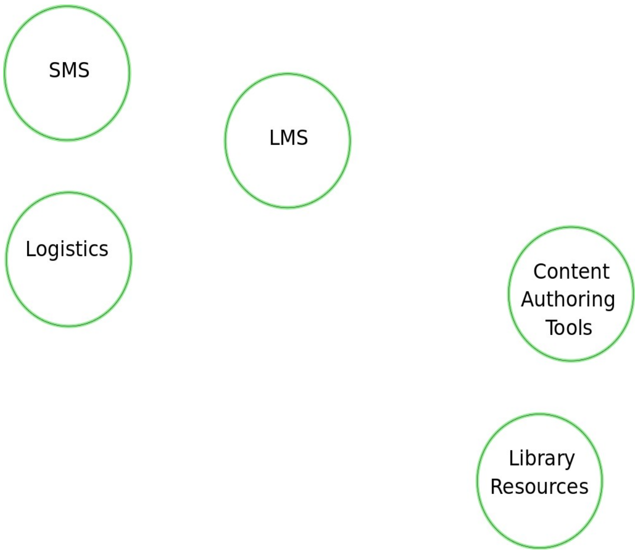

Figure 6.1: Early deployed Educational Technology systems without significant integration.

2. Early integrations, more systems deployed

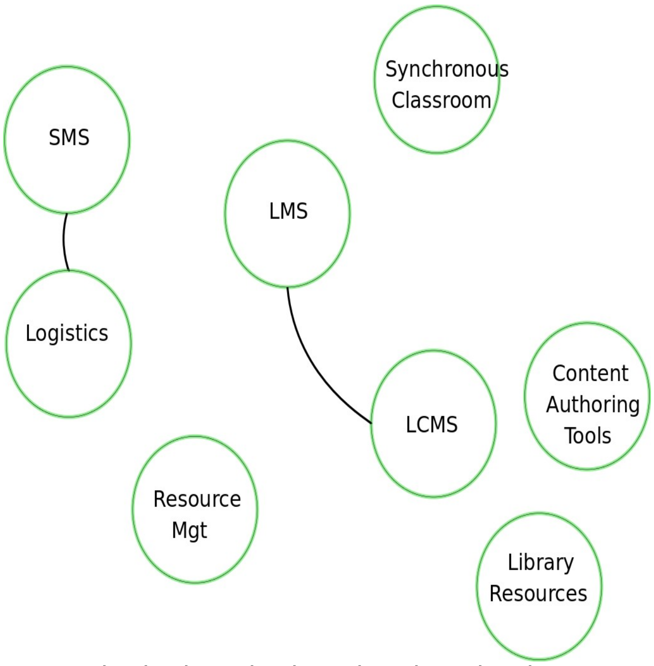

Figure 6.2: Early deployed Educational Technology systems with early integrations.

3. Systems proliferate, integrations more dense, usability poor due to the need to login to multiple systems usually with multiple user account details.

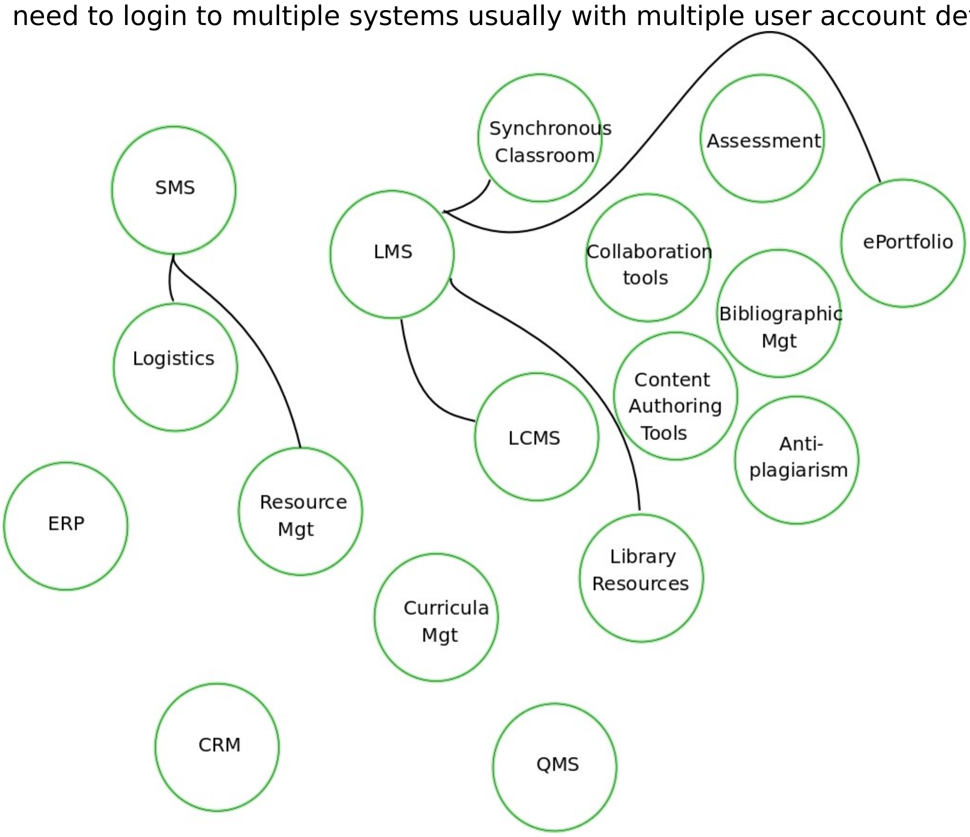

Figure 6.3: Fuller systems deployment, fuller integration but still multiple system access required

Improved identity management is possible in the above architecture by creating Single Sign On (SSO) access to the various systems. This is a common approach and uses technologies such as LDAP, Kerberos, Shibboleth, etc to achieve it.

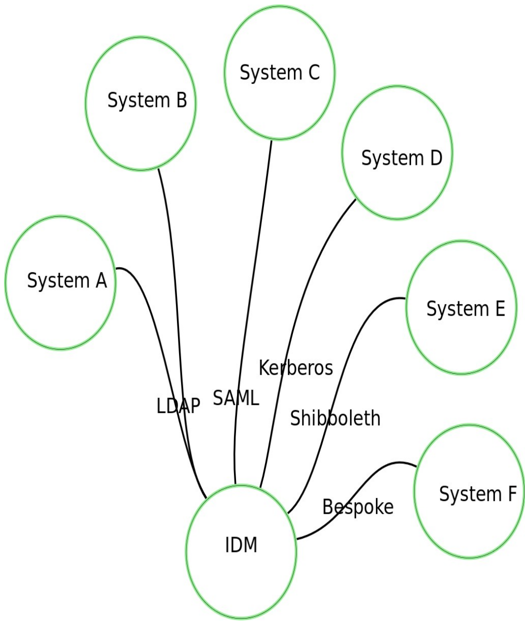

Figure 6.4: Example of SSO integration using common SSO technologies.

While this can eliminate multiple user account issues, it does not serve to integrate the key data in the other systems. For example, SSO to the LMS does nothing to manage the enrolment business logics ultimately controlled by the SMS.

4. Full systems deployment, more complete integration and usability managed through 'portalised' access

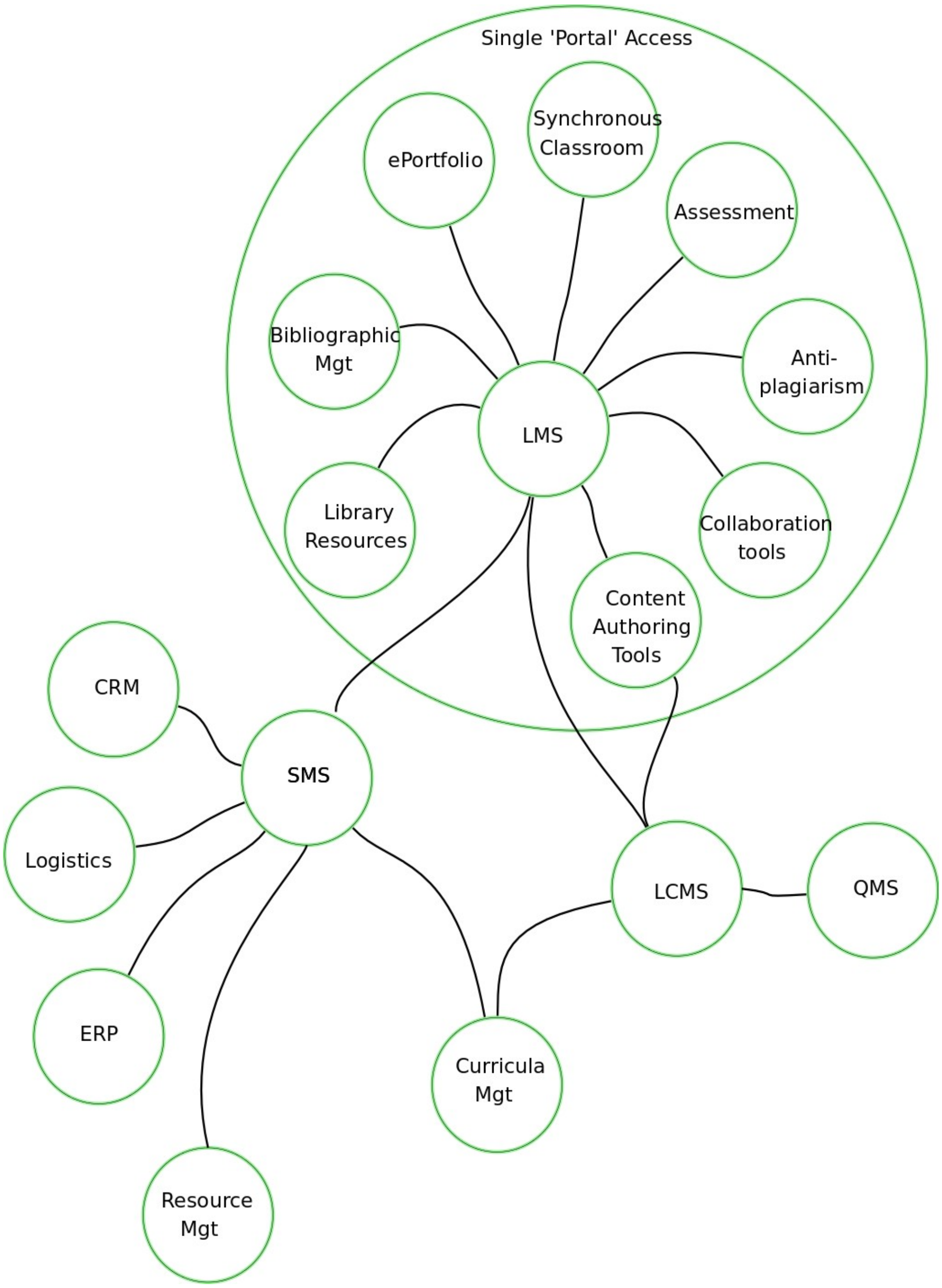

Figure 6.4: Full systems deployment with fine-grain integration and 'Portal' user access for better usability.

7. Sample integration technologies

There are a range of technology standards available for inter-system integration. These vary by system and the business domains over which the systems interoperate. For example integration standards used with library systems include the now aged z39.50 protocol which is maintained by the US Library of Congress and uses ANSI/NISO standard Z39.50, and ISO standard 23950. This has been perhaps superseded by the more modern Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH). There is also the Metadata Encoding and Transmission Standard (METS) which is another metadata standard for describing objects within a digital library.

In the area of learning content and learning content delivery systems there is the Shareable Content Object Reference Model (SCORM) managed by non-profit Advanced Distributed Learning set up originally by the US Department of Defence. The SCORM is an umbrella of standards-based specifications including:

- Content Aggregation Model: the IMS standards for metadata and content packaging including the SCORM IMS Question Test Interoperability (QTI) standard for encapsulating assessments.
- Run-Time Environment: the SCORM Shareable Content Object (SCO) JavaScript framework for content interoperation and user data persistence in an LMS database when launched from within an LMS containing a conformant SCORM application programming interface (API) adaptor.
- Sequencing and Navigation: scope for multiple learning activity pathway mapping and enablement.

For user authentication there are a myriad of standards available including Lightweight Directory Access Protocol (LDAP), Kerberos, Shibboleth, Security Assertion Markup Language (SAML) and so on.

There are also more generic integration technologies available such as Extensible Markup Language Remote Procedure Call (XMLRPC) and SOAP (originally 'Simple Object Access Protocol). Similarly there is the use of Public Key Infrastructure (PKI) to establish and authenticate using an encrypted public and private identity key pair. Less robust but similar is the use of domain or IP address based trusted referral where for example requests fro URLs identified as being made from within an trusted domain or from a trusted IP address or range of addresses can be treated different;y to other access.

Finally, although they cannot really be considered as standards, there are formats that can be used in flat-file transfers such as Comma Separated Values (CSV) and so on.

So there is a veritable plethora of choice when considering integrations across systems in order to produce a more seamless user experience and leverage the powers of the constituent systems over a broader sphere. The challenge is choosing wisely and implementing robust integration interfaces. To be able to choose wisely requires an understanding of the individual technical protocols combined with a broader architectural view of the entire systems deployed plus a understanding of the broad business rules to be addressed.

Here then is an example of a broadly integrated architecture with common integration technologies mapped:

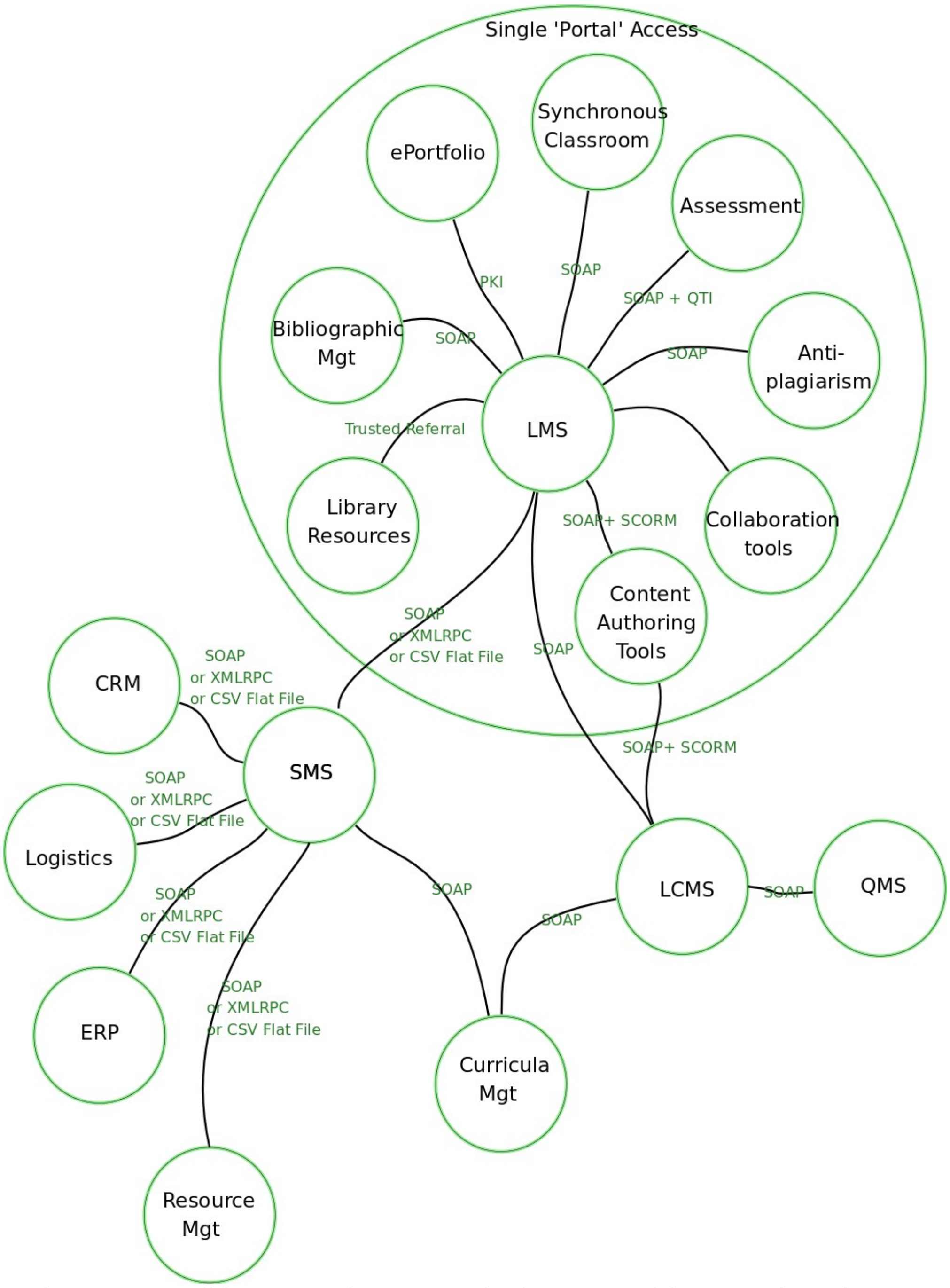

Figure 7.1: Full integrated systems deployment with examples of common integration technologies used.

8. A look to the future: recent impacts of Content Management and content federation on architecture

There are several, recent architectural opportunities emerging within the industry that can have a profound impact on the way Educational Technology systems are integrated, specifically from a content interoperation perspective. These opportunities stem from the maturation of the LCMS into a more powerful system, the development of improved content interoperability standards and in particular standards for federation of content between repositories. The opportunities also stem from the serendipitous, commoditised separation of the content management from presentation systems such as the LMS. The architectural outcomes of these opportunities serve as an excellent exemplar of the vector in which extensible system integration is progressing and give a glimpse of the road ahead.

State-of-the-art LCMSs such as Equella can not only manage version control, content state, workflow, collaboration but can also effectively switchboard content from collection internal to the LCMS and collections beyond or 'upstream' via content federation. Access controls on content can be mapped to target presentation system, user, group, role, network topology, content state, digital rights an defined against functional capabilities (search, discover, edit, clone, publish, tag, etc).

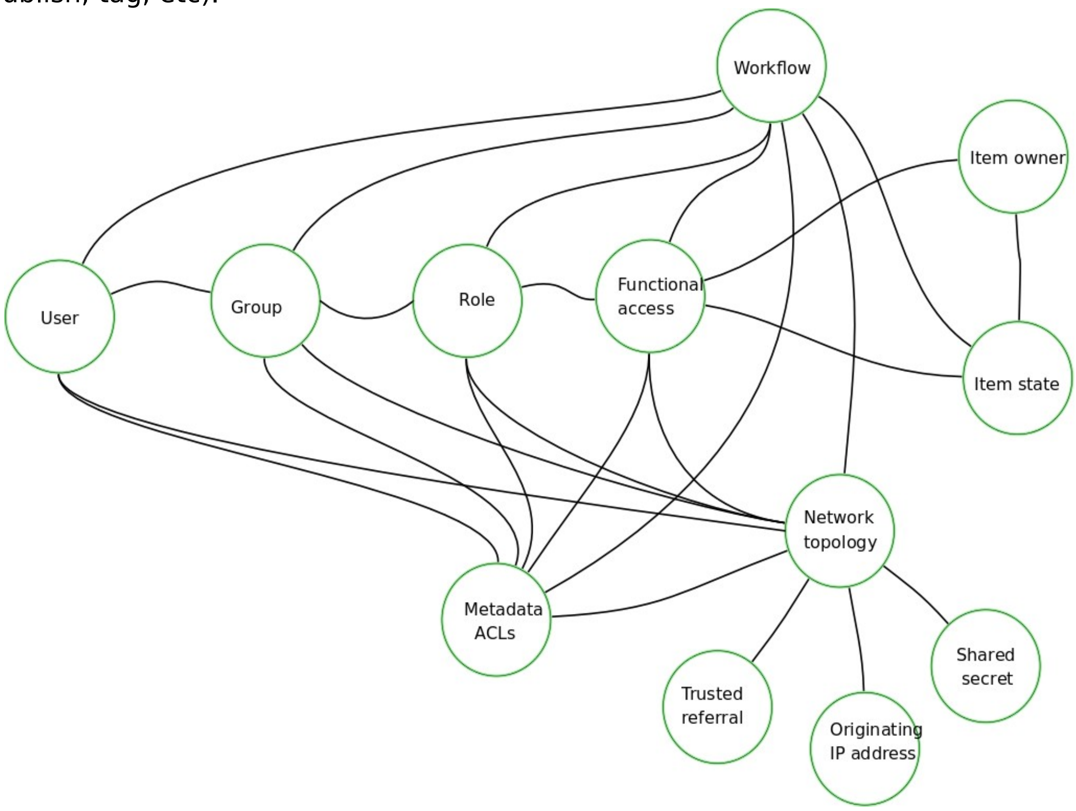

Figure 8.1: Sample content-user access control model – taken from the capabilities of the Equella LCMS product.

In addition to this 'content switchboard' capability, the close integration of the LCMS with presentation systems such as, for example, the LMS, means that content can be seamlessly discovered, edited, aggregated and published from the LCMS into the LMS. When this is combined with the ability to access both internal and external collections via the LCMS, the possibilities of federation of content change profoundly from 'federate and discover' and, perhaps, 'view' to 'federate, discover, aggregate, localise, publish and use'. This content 'switchboard' model opens up huge possibility in terms of an integrated content platform as a key component in a good Educational Technology Systems Architecture.

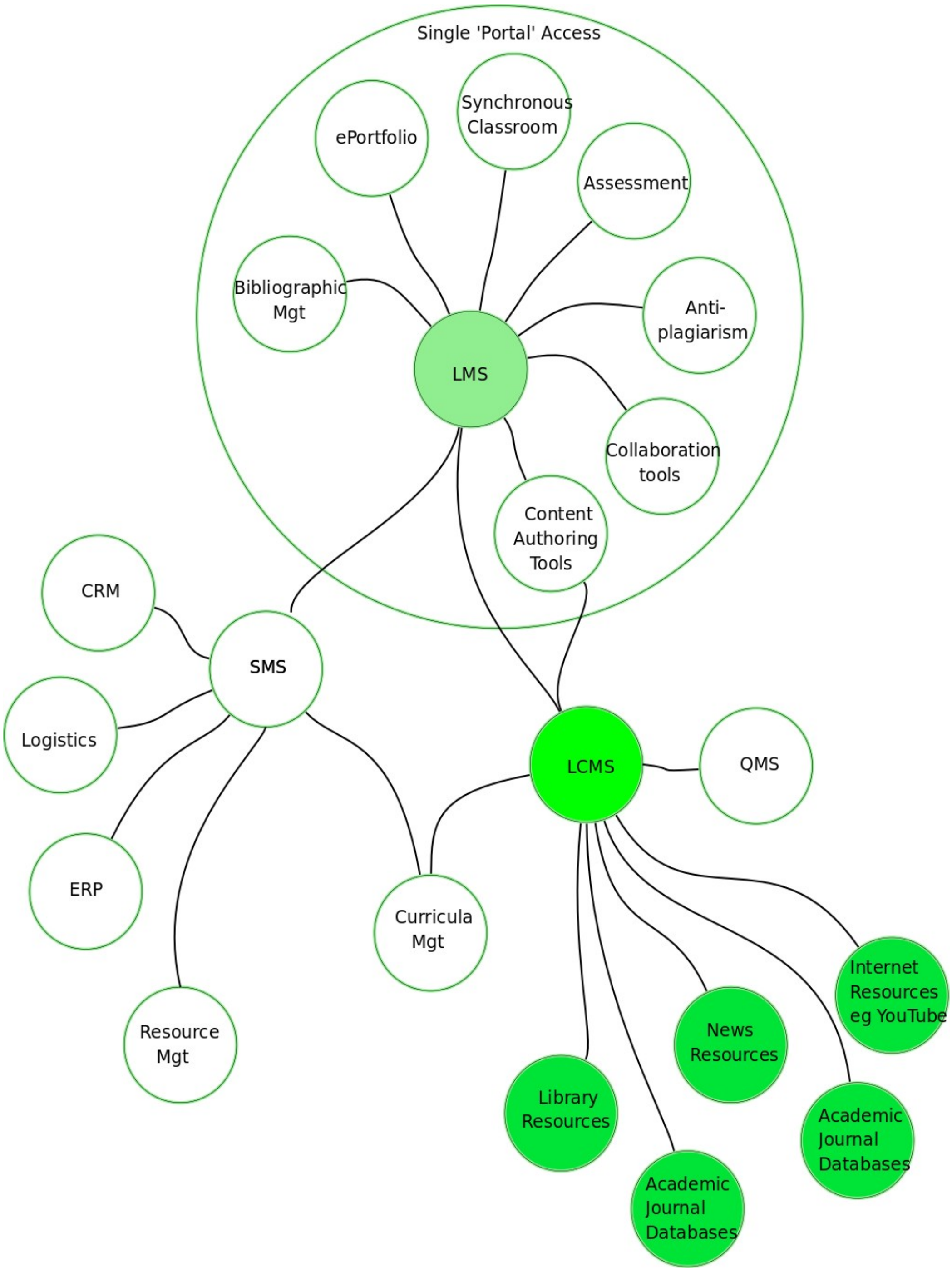

Figure 8.2: Federate and consume content capability model via the LCMS as 'content switchboard'.

This model can be extended as the LCMS integration is extended into other presentation systems, effectively making an invisible by extremely powerful content management layer beneath all content consuming systems in the architecture yielding great advantages of content discoverability, reuse and management.

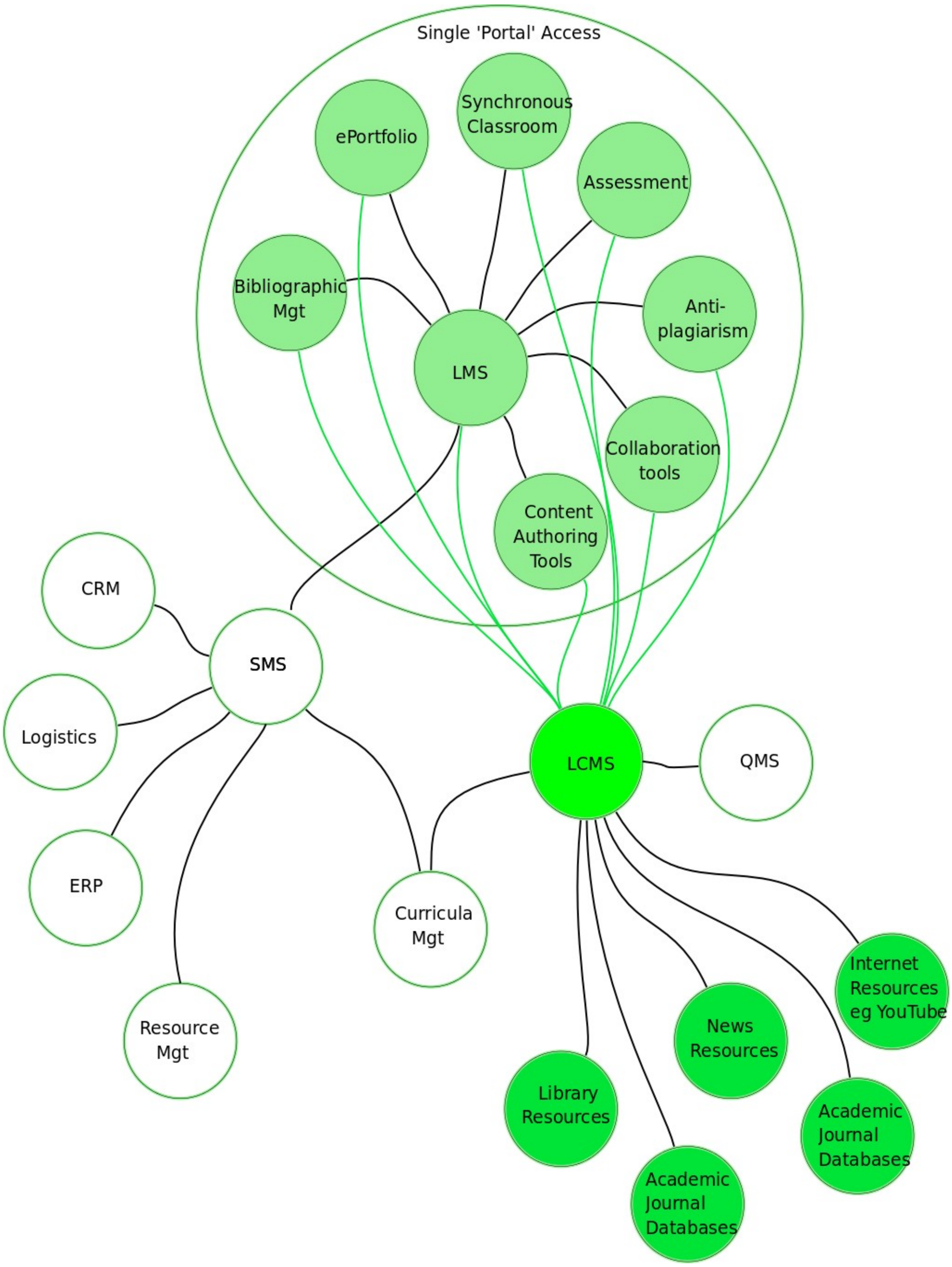

Figure 8.2: Federate and consume content capability model via the LCMS as content 'switchboard' to all front-end systems.

This 'content switchboard' model extends further however. The architecture examined thus far has assumed a single teaching and learning or 'front end' set of systems and this has been presented in best-practice format as a single access or 'portalised' structure. Many Educational institutions support multiple teaching and learning portals however. For example many universities have different front-end systems for their faculties, often with different branding and often with different component applications. Teaching and learning systems may also be diversified to separate access and control. For this reason for example K12 schools frequently have separate LMSs, effectively separating different school cohorts and their teaching and administrative staff. The use of the LCMS as a content switchboard component in an Educational Technology Systems Architecture can serve in this context extremely well.

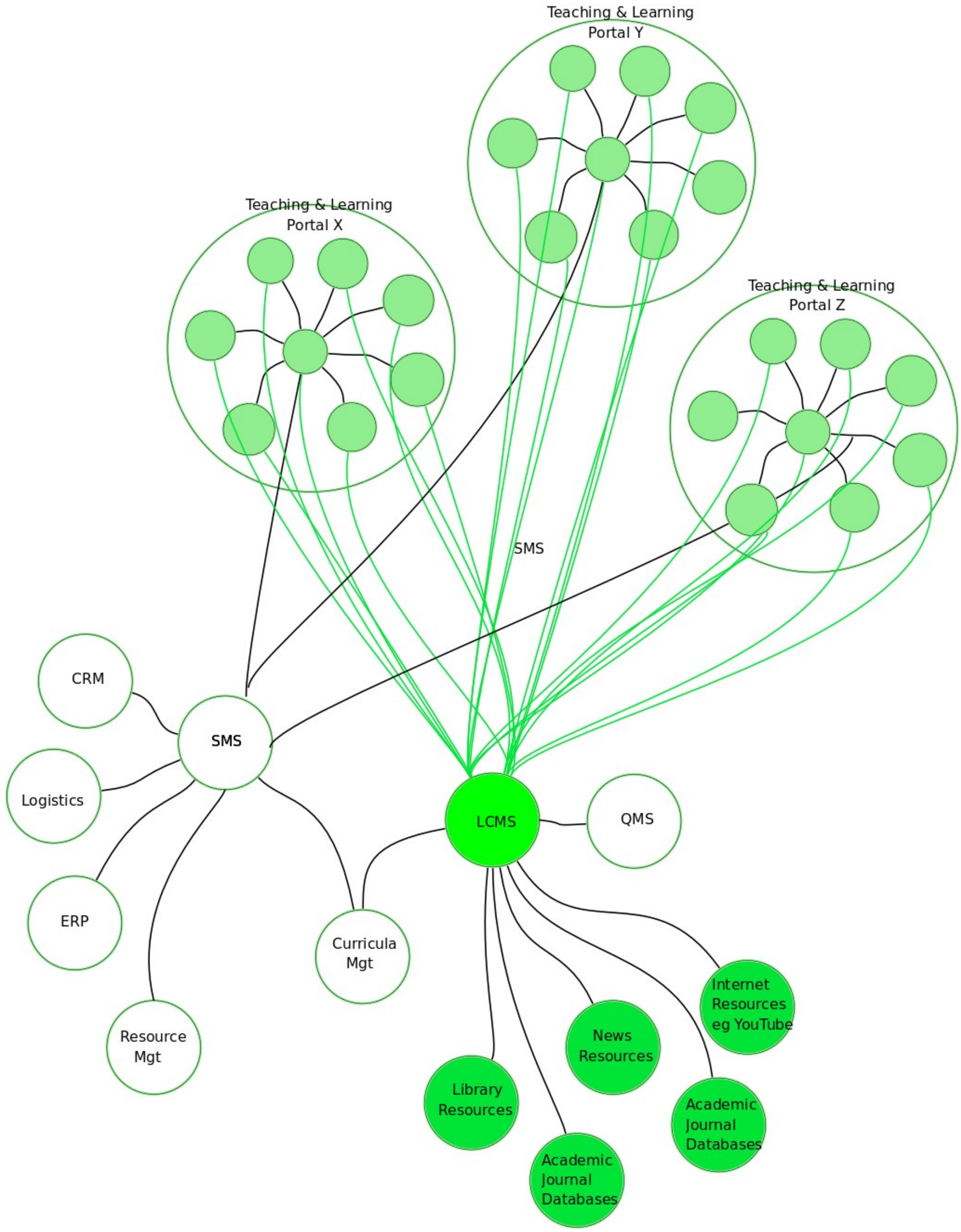

Figure 9.1: Multiple Teaching-Learning Portals with one content switchboard – an intra-institutional content access model.

This architecture is extensible to the inter-institutional level. It again involves multiple Teaching-Learning Portals but also involves multiple institutional LCMSs.

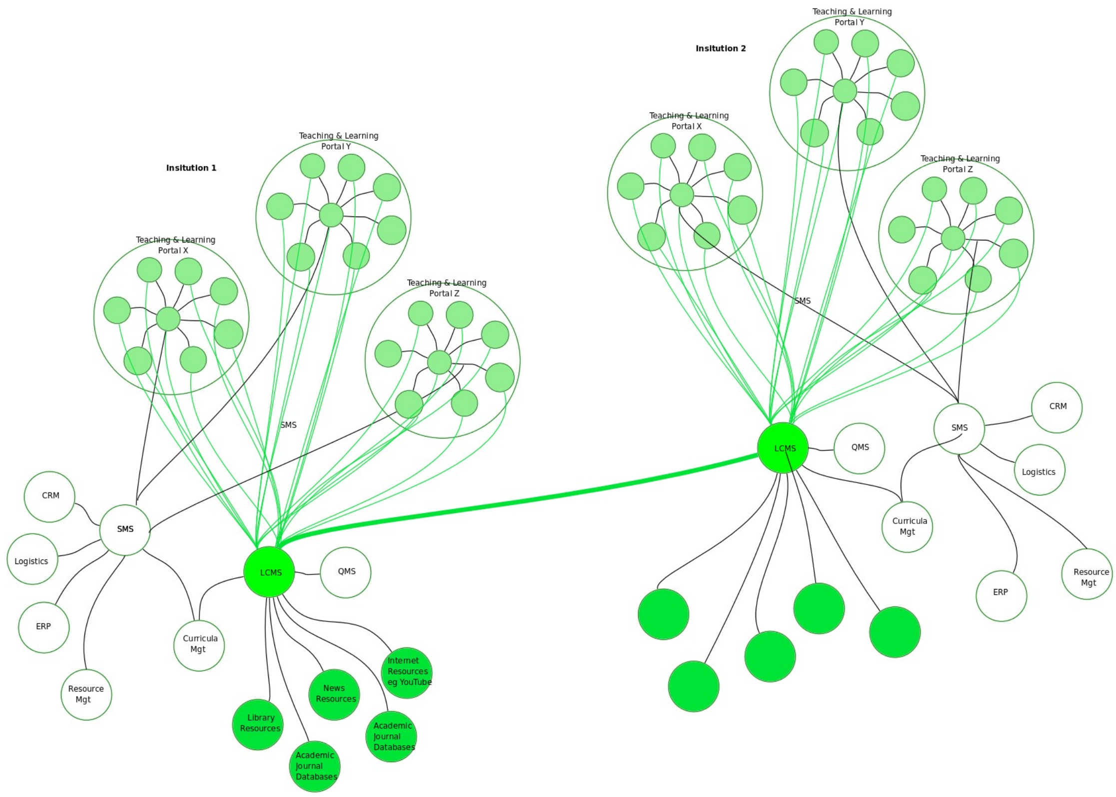

Figure 9.1: Multiple institutions accessing content via multiple LCMS switchboards – an inter-institutional model.

9. Conclusions

Firstly, It is fair to conclude that conceiving, designing and implementing a good Educational Technology Systems Architecture to robustly and efficiently serve the needs of the process of teaching and learning in an Education institution is hard. It involves disciplines that are too new to have seen received expertise disseminated into the Education industry. It includes huge complexity in terms of systems, technologies and reflects diverse business processes spanning business units right across an implementing institution. However, like any complex and difficult process, understanding how the challenges can be broken down and addressed is the first stage to making it a simpler and more reliable and cost effective process. Each of the pieces of the puzzle can be addressed. The key challenge is for institutional leaders to ensure that they are and that they take a strategic view of their Educational Technology systems and how the interoperate. They need to be informed and apply realistic resources to the process of making the systems interoperate effectively and they generally need to do it better than they are today.

Secondly, despite the dramatically changing architectural landscape emerging as systems have proliferated, it is simple to extrapolate into a longer historical context and conclude that the industry is still very close to the beginning of what is clearly a long journey. The complexity of what lies even most immediately ahead in content platform possibilities dwarfs what lies behind. However it is also reasonable to expect that the standards and technologies and developed integration disciplines and expertise that lie ahead will simplify the problem space.

Finally, stepping beyond the challenges, the radical possibilities emerging in the area of content interoperation within a broad Educational Technology Systems

Architecture show that the opportunities for growth are immense. This also suggests that like in any technological advancement, some institutions will be left far behind while others surge ahead. This 'caterpillar effect' is likely to impact broader and core education outcomes such as educational quality and compliance not to mention cost. The key is to recognise the evolutionary path and to actively pursue it both strategically and tactically. To fail to do so is almost certainly to fail more broadly.