

Архитектура и инфраструктура на университетска информационна система

Мирослав Михайлов, Цветелин Павлов

Architecture and infrastructure of a university information system: This paper revue the architecture and infrastructure design of a university information system.

Universities are complex organizations which run significant number of strongly correlated processes. Universities' environments are dynamic and regulations could change frequently. Existing information systems usually fail to cover all processes and to serve them efficiently and do not offer an adequate solution of the problems from the specific domain. Most often this is due to conceptual errors in the architecture design. In search of a solution to these problems a new architecture, based on modern concepts was designed and corresponding infrastructure to host the development process was built.

Key words: university information system, architecture, service-oriented architecture, enterprise service bus, infrastructure, data center.

ВЪВЕДЕНИЕ

В университетите като институция се преплитат сложни взаимовръзки между различни групи хора – студенти, преподаватели, научни сътрудници, академично ръководство, частен бизнес. Съществуващите университетски информационни системи (УИС) най-често са разработени от различни колективи в динамично променяща се среда от изисквания, което води до значителната им несъвместимост, несигурност и непълна функционалност. В същото време потребителите на групите потребители в университетите имат интерес да изпълняват дейностите си без усложнения и за възможно най-кратък период от време. Ето защо, наличието на съвременна информационна система, която да обезпечава управлението на работните процеси в университета, да подпомага взимането на управленски решения, както и да автоматизира и оптимизира работата на потребителите, е критичен фактор за поддържане на организацията в нормален работен ритъм.

За да изпълни тези цели ефективно, информационната система трябва да е проектирана гъвкава, мащабируема, сигурна, надеждна, трябва да е съобразена с процесите в организацията и да изпълнява индустриално утвърдените добри практики за построяване на информационни системи от този клас. Всички тези изисквания правят проектирането на архитектурата и функционалната инфраструктура, в която ще бъде внедрена системата изключително критично за успеха на софтуерния проект.

АНАЛИЗ НА ПРЕДМЕТНАТА ОБЛАСТ

Образователната система в сферата на висшето образование има уникална характеристика, която прави моделирането трудно, а адаптирането на софтуерни решения от други области – неефективно или практически невъзможно: потребителите на системата могат да участват в множество роли едновременно на еднакви или различни нива. Например, един студент може да участва едновременно в няколко специалности за различни ОКС, в научни проекти или да е част от административния апарат на университета и др. Тази характеристика поставя високи изисквания за гъвкавост към архитектурата, както и за функционалности, даващи възможност за моделиране на работните процеси по време на експлоатация на системата.

В предметната област влизат три типа потребителски роли: „обучаващи се“, „обучаващи“ и „административен персонал“ (последната включва и академичното ръководство). Необходим е механизъм за причисляване на един потребител към една или няколко потребителски роли, като едновременно с това той получава всички разрешения и забрани, функционалности и участие в работни процеси,

определени за съответната роля.

Ролите от своя страна могат да участват в определени единици от структурата на университета. Архитектурата трябва да осигури механизъм за назначаване на привилегии и функционалности според принадлежността към определена структурна единица.

Работните процеси в предметната област могат да се разделят главно на „административни процеси“ и „образователни процеси“. Отново, един и същ потребител може да участва в двата вида процеси. Архитектурата трябва да осигури гъвкав механизъм за обвързване на работните процеси с потребителски роли и конкретни потребители, като отчита тази характеристика на предметната област.

Преплитачите се различни интереси могат да доведат до нерегламентирана употреба или злоупотреба. Системата трябва да осигури мощни възможности за удостоверяване на самоличността, проверка на правата за достъп и моделиране на сигурността въобще.

От направения анализ на предметната област може да се направят следните изводи с пряко отношение към архитектурата на УИС:

- ✓ Изграждането на всички функционалности в една монолитна система може да доведе до усложнена експлоатация на решението заради размитите граници между ролите и сложните зависимости между работните процеси. Необходимо е интегрирано решение от множество подсистеми, всяка от които решава проблем в определена и обособена част от предметната област.
- ✓ Силните зависимости между процесите и работните обекти могат да доведат до „обектна експлозия“ при проектиране, както и до т.нар. „спагети ефект“ и сложни регресивни грешки при промяна или корекция на изходния код. Затова е необходима слаба обвързаност между интегрираните подсистеми, каквато осигуряват сервисно-ориентираните или „облачните“ архитектури.
- ✓ Заради силните зависимости между процесите, размитите граници между потребителските роли и честата промяна на нормативната уредба, архитектурата трябва да позволява моделиране на работните процеси по време на експлоатация. Подобно решение предлагат архитектурите с корпоративна сервисна шина.
- ✓ Тъй като университетската информационна система обслужва и критични за университета работни процеси, необходимо е да се осигурят условия за продължаваща интеграция включително и с наследените системи. По време на експлоатация е вероятно да се наложи интеграция със системи на други организации (МОМН, други университети) и интеграция с вече внедрени системи на други производители (счетоводни, за документооборота). Такова решение предлагат клас софтуерни продукти, наречени интеграционни сървъри.
- ✓ За да се осигури стабилност и надеждност при експлоатация, както и за да се елиминират рисковете за „напрежение“ в точките на интеграция, е необходимо архитектурата да се изгради върху хомогенна платформа, като се използват продукти от един производител.
- ✓ За да се осигурят добри възможности за мащабируемост е необходимо използването на корпоративен клас продукти.
- ✓ За да се постигне максимална достижимост, навременност и правилна насоченост на информацията към отделните потребители, е необходимо архитектурата да е хетерогенна от гледна точка на използваната технология за клиентските приложения, т.е. архитектурата не трябва да налага ограничения при избора на технология.

АНАЛИЗ НА РАБОТНАТА СРЕДА

Правилната преценка при проектиране на архитектурата на една информационна система изисква и анализ на работната среда, като се направят съответните изводи за точките на интеграция с наследените системи, апаратните и експертни ресурси, с които разполага организацията и усилията за поддържане в изправно състояние, които ще са необходими.

Русенският университет, както и всички университети, разгледани в хода на проучванията за проекта за УИС:

- ✓ разполагат със специализирани структурни звена от експертен персонал, способен да обслужва информационна система от по-висок клас, но е възможно да се наложи допълнително обучение;
- ✓ имат, макар и вариращи, експертни и финансови възможности за изграждане и поддръжка на системен център;
- ✓ имат изградена комуникационна инфраструктура, компютърна мрежа и са осигурени работни станции за преподавателите, служителите и студентите, както и възможности за използване на личен компютър;
- ✓ От анализа на работната среда могат да се направят следните изводи:
- ✓ Необходима е внимателна преценка за технологията, която се използва за изграждане на клиентските приложения, съобразена с утвърдените добри практики за изграждане на информационни системи.
- ✓ Връзката между нивата на информационната система трябва да става чрез стандартизирани или утвърдени от индустрията протоколи. Най-подходящи за почти всички случаи на употреба са протоколите за уеб услуги.

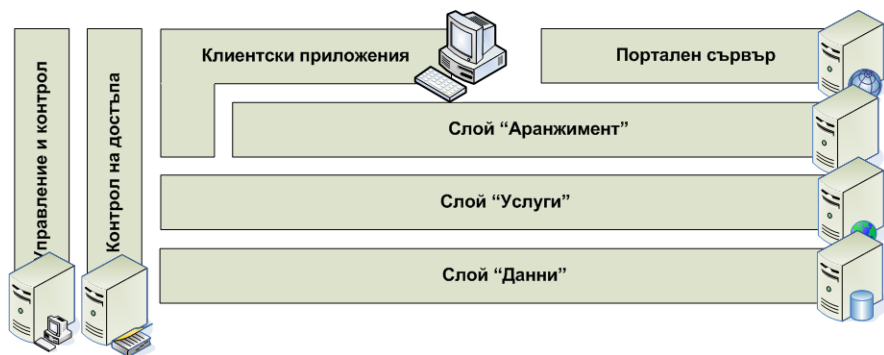
АРХИТЕКТУРА

На базата на направения анализ на проблемната област и средата за университетска информационна система е проектирана сервизно-ориентирана архитектура, базирана на корпоративна шина за услуги.

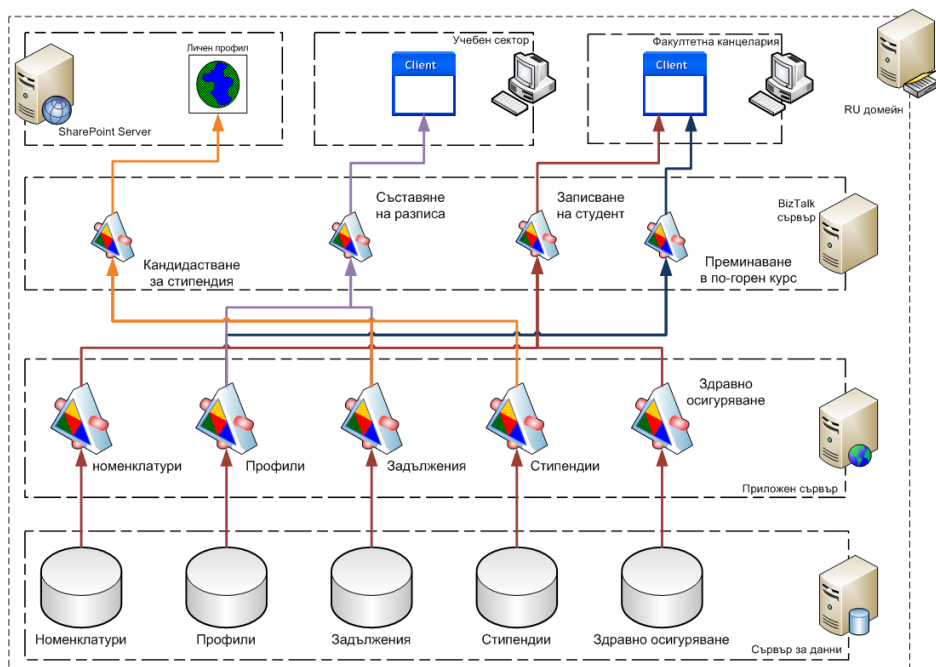
За платформа е избрана Microsoft .NET, версии над 3.0, заради подобрените възможности за разработка на многослойни системи. Изборът на тази платформа и на продукти от същия производител спомагат за изграждане на хомогенна архитектура, намаляват рисковете от грешки в точките на интеграция или ги извеждат на позиции, където са добре известни и очаквани, спомагат за организиране на удостоверяване на самоличността и контрол на достъпа, изнесени в операционните системи под информационната система, както и осигуряват възможности за последващо развитие на системния център, разширяване на функционалностите и повишаване на капацитетите при необходимост.

Архитектурата обхваща четири вертикални слоя (Фиг. 1). В слой „Аранжмент” се моделират работните процеси, той изгражда корпоративната шина за услуги, и осигурява точки за интеграция със системи на други организации или с наследени системи, като близостта му с клиентския слой спомага за по-добра производителност в сценариите на такава интеграция.

Клиентският слой е разделен на две части – десктоп приложения и уеб-базирани приложения („портален сървър” на Фиг. 1), като типа на клиентското приложение се избира според конкретните изисквания към него и утвърдените практики при разработка на приложения.



Фиг. 1 - Диаграма на слоевете



Фиг. 2 - Компоненти на архитектурата по слоеве и взаимодействие между тях

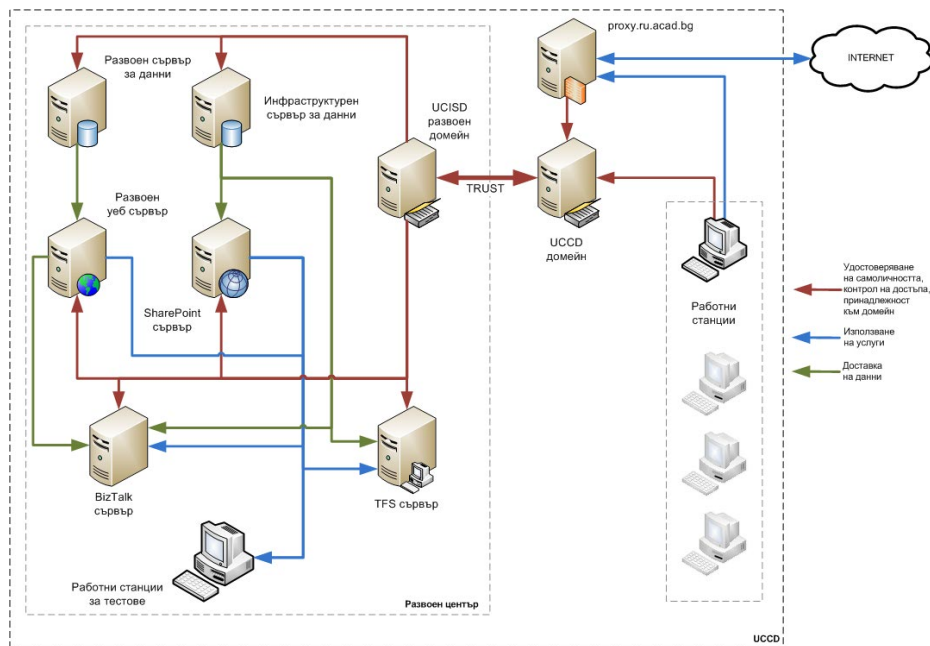
На Фиг. 2 е показано разпределението на отделните компоненти на информационната система по слоеве и взаимодействието между тях. Включени са само няколко от интегрираните информационни системи и работни процеси.

Добър пример за илюстрация е процесът „кандидатстване за стипендия“. Първата стъпка от този процес е проверка за изпълнени задължения от страна на студента (заверен семестър, взети изпити, среден успех по-висок от минималния). Ако студентът отговаря на поставените условия, процесът трябва да го регистрира за участие в класиране и да изчисли състезателния бал. Регистрацията става чрез използване на услугите за стипендии, като данните се запазват в базата данни за стипендии. Проверката за задълженията става чрез обръщение към услугите за задължения, които използват информация от базата данни на подсистемата.

Точката на интеграция между двете системи остава в слой „Аранжмент“, а процесът може да бъде променен по време на експлоатация на интегрираната система без да се налага корекция на изходния код.

ИНФРАСТРУКТУРА

Заради мащабите на системата, продължителната разработка, изискванията за надеждност, стабилност и продължаваща интеграция, инфраструктурата включва два центъра – „развоен“ и „системен“. Първият е необходим за процеса на разработка, а вторият – за процеса на експлоатация на завършената система. Така се елиминира евентуален риск процесът на разработка да повлияе на нормалната работа на вече внедрените системи и се гарантира тяхната стабилност. Подсистемите се внедряват в системния център само след като са преминали всички тестове и са във финалния си стадий.



Фиг. 3 - Топология на развоен център

Развойният център е обособен чрез изграждане на собствен домейн, базиран на Active Directory Services, който е в „отношение на доверие“ с използвания в Русенския университет домейн.

Развойният център включва два сървъра за бази данни – един, използван от компонентите на инфраструктурата и един използван за разработваната система, за да не може развойната дейност да причини авария в инфраструктурата на центъра.

В развойният център е внедрена система за екипна работа, която включва система за контрол на изходния код. Системата за контрол на изходния код е интегрирано решение, което позволява да се запазват версии на изходния код, да се правят разклонения на кодовата база, да се следят промените на кода и да се работи едновременно от няколко члена на екипа върху един негов елемент. Към нея е включен и инструмент за автоматизирано провеждане на тестове. Системата за

екипна работа позволява назначаване на задачи, проследяване на грешки и изисквания за качествен контрол.

Останалата част от центъра трябва да осигури точки за разположение на архитектурата в инфраструктурата на развойния център и условия за разработка на системи, които наподобяват максимално очакваните условия в системния център. Това включва уеб сървър, интеграционен сървър и портален сървър. Уеб сървърът се използва за уеб услугите от слоя „Услуги“. Интеграционният сървър изгражда корпоративната шина за услуги и отговаря на слоя „Аранжмент“, а порталният сървър осигурява функционалности за изграждане на уеб-базирани клиентски приложения. Освен сървърите, в домейна на развойния център се включват и набор от виртуални машини за симулация и провеждане на тестове на разработените клиентски приложения.

Системният център се различава предимно по това, че не включва инструменти за разработка и отговаря на повишени изисквания за надеждност, сигурност и мащабируемост. Това включва възможности за изграждане на кворум от сървъри за бази данни и ферми от портални и интеграционни сървъри, както и хардуерно оборудване от по-висок клас.

ЗАКЛЮЧЕНИЕ

При проектирането на архитектура за УИС са отчетени особеностите на предметната област и работната среда, използвани са съвременни технологии, методи и подходи, благодарение на които се решават нерешени до сега проблеми и се осигурява максимално подравняване на реализацията с изискванията на университетите.

Проектираната и изградена инфраструктура е съобразена с изискванията на архитектурата и се използва активно в процеса на разработка на университетската информационна система.

ЛИТЕРАТУРА

[1] Маринов, М. Информационни системи - анализ и проектиране. Русе : ПРИМАКС, 2005. стр. 160. Том. второ, преработено издание. ISBN 954-9972-62-3.

[2] Microsoft Patterns & Practices. Application Architecture for .NET: Designing Applications and Services. 2002 г.

[3] Microsoft Patterns & Practices. Designing Data Tier Components and Passing Data Through Tiers. 2002 г.

[4] Joseph, Jossy, и др. Architectural Patterns for Distributed Computing. The Architecture Journal. октомври 2008 г., 17, стр. 6-13.

[5] Microsoft Corporation. Enterprise Service Bus Guidance. Codeplex. [Онлайн] [Цитирано на: 18 октомври 2009 г.] <http://www.codeplex.com/esb>.

За контакти:

доц. д-р Мирослав Михайлов, директор на ЦИКО, Русенски университет “Ангел Кънчев”, тел.: 082 888 782, e-mail: mmihaylov@ru.acad.bg.

маг. инж. Цветелин Павлов, ЦИКО, Русенски университет “Ангел Кънчев”, тел.: 082 888 328, e-mail: cpavlov@ru.acad.bg.

Публикацията е реализирана по проект по ФНИ РУ-05

Докладът е рецензиран.