

SpiderCloud – Облачна реализация на среда за Control Network Programming

Цанко Големанов, Костадин Крачанов, Емилия Големанова

Control Network Programming (CNP) is a style of high-level programming that is especially effective for solving problems that have natural graph-like representation of imperative, declarative, or mixed nature. SPIDER is a CNP-language. The report is aimed to present an approach of CNP online-compiler design and the implementation of CNP in Cloud environment. The main features of IntraWeb-technology are shown. A web application for SPIDER-language programming, modifying and compiling is described.

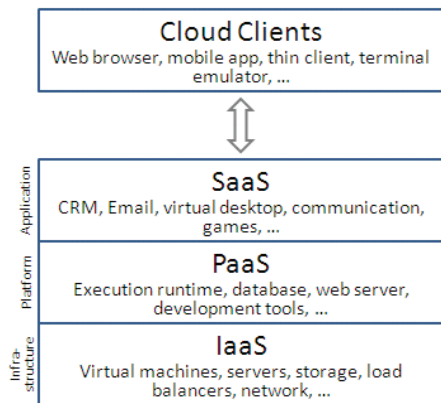
Key words: Control Network Programming, CNP, SpiderCNP, Online IDE.

ВЪВЕДЕНИЕ

Понятието Cloud Computing (Изчисления в Облак), абсолютно непознато до преди няколко години, вече се налага като една от основните насоки в развитието на информационните технологии [1]. В основата му стои идеята локалните изчислителни продукти и средства да бъдат заменени от предоставяни отдалечени такива. Новият модел на компютърни технологии получава названието “Облачни компютърни технологии”, като облакът е метафора за интернет – т.е. технологии и услуги, достъпни през глобалната мрежа. Моделът позволява сериозни по сложност и обем изчислителни софтуерни продукти, информационни масиви и структури да станат достъпни за потребители през елементарен web-browser. В доклада ще бъдат разгледани подходите и технологиите, с помощта на които програмирането чрез Управляващи мрежи [2] (езикът SPIDER) е пренесено и в Облачното пространство.

Характеристики и особености на Облачните технологии

Основен принцип на Облачните технологии е „Всичко е услуга”. Според това разбиране, на отдалечените потребители (Cloud Clients) се предоставят за съвместно използване (фиг.1) три основни модела на услуги: IaaS, PaaS и SaaS.



Фиг. 1 - Модели Облачни услуги

- **Софтуер като услуга** - Software as a Service (**SaaS**) – това е най-често използвания бизнес-модел, при който на потребителите се предоставя отдалечен достъп до конкретна софтуерна система (пакет) или база данни. Така се реализира модел на използване на „софтуер по заявка”, а заплащането е за конкретно време или за обем на информационния трафик.

- **Инфраструктура като услуга** - Infrastructure as a Service (**IaaS**) – това е най-базовия модел на облачна услуга, при който на потребителите се предоставят за използване компютърни системи – както физически така и виртуални. В тази инфраструктура потребителите инсталират операционните системи и приложния софтуер, който им е необходим;
- **Платформа като услуга** - Platform as a Service (**PaaS**) – при този модел потребителите получават изградена изчислителна среда, включваща операционна система, различни IDE, DB-сървъри, WEB-сървъри и др.

Предимствата от използването на Облачни технологии могат да се локализируют в различни направления, сред които са: **минимизиране на разходите** за инвестиции и оперативна дейност (поддръжка, енергия и др.); **плащане при използване** – потребителите плащат само това, което използват, а използват само когато услугата им трябва; **гъвкавост и еластичност** при мащабирането на необходимите ресурси; **отдалечен и мобилен достъп**; **зелени технологии** – постига по-висока степен на оползотворяване на хардуера и в резултат се изразходва по-малко енергия; **автоматизация** – осигурена е възможност за изграждане, инсталиране, конфигуриране и доставяне/използване на приложения и услуги само през интернет без необходимост от ръчна намеса.

Въпреки, че отношението към Облачните технологии не е еднозначно [3], новата изчислителна технология намира все по-голямо разпространение, работи се по стандартизацията ѝ [4] и преобладаващо е мнението на специалистите, че ще е основен приоритет през следващите няколко години [5].

Докато IDE SpiderCNP [6] може безпроблемно да се използва както локално, така и в произволна PaaS-инсталация, то представяната среда SpiderCloud е конкретно настроена да работи в SaaS-модел.

Дизайн на средата SpiderCloud чрез технологията IntraWeb

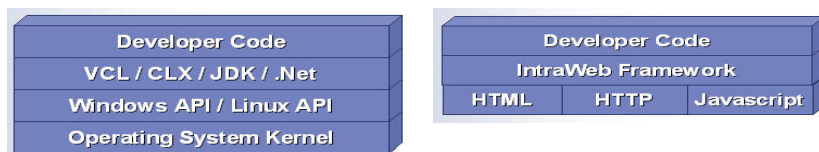
Основното предназначение на IDE SpiderCloud (в текущия ѝ вид) е да реализира начално и базово запознаване с принципите на програмирането чрез Управляващи мрежи. За целта е необходимо да се осигури лесен и бърз достъп до средата, без необходимост от закупуване, инсталиране и конфигуриране на сложни съпътстващи продукти и системи. Стартовата помощна информация за използването ѝ също трябва да бъде съвсем минимизирана. Разбира се това води до сериозно оръязване на функционалността на средата, за сметка на простота, лекота и бързината при използването ѝ. В този контекст SpiderCloud се включва в класа системи известни като Online compilers, за които са характерни:

- Лесен и бърз интернет-достъп – SaaS модел;
- Зареждане или разработка на нови конзолни (текстови) приложения на избрания език за програмиране;
- Опростен потребителски интерфейс;
- Компилиране и възможност за стартиране и/или download на генерирания изпълним обектен код.

За реализация на IDE SpiderCloud като web-базирано приложение бе избрана авангардната технология IntraWeb [7], разработвана и предлагана от компанията Atozed Software. IntraWeb представлява инструмент за създаване на weblications – web-приложения с интерфейс и функционалност аналогични на локално-стартираните. Изграждането на weblication е много по-различно от изграждане на Web-сайт, а IntraWeb е първият инструмент, специално насочен към създаване на weblication-и. Използването му позволява разработването на сложни Web-приложения да става изключително бързо и лесно. Докато в типичното Web-приложение разработчикът трябва детайлно да управлява комуникацията с много различни протоколи от ниски нива и стандарти като HTTP, Javascript и HTML, то при типичното IntraWeb-приложение изобщо не му се налага да се занимава с това.

При разработката на локални приложения .NET Framework, VCL (Borland Tools), CLX (Kylix), и JDK (Java) осигурят на разработчиците едно високо ниво на абстракция, известно като "Class Frameworks". То изолира разработчика от долното API-ниво на операционната система и му гарантира необходимия комфорт за бърза и безпроблемна работа. В редки случаи, ако е необходимо, разработчикът може да заобикаля "Class Frameworks" и да се обръща директно към функциите на API.

IntraWeb функционира по подобна схема. Тук HTTP, HTML и Javascript играят ролята на API от ниско ниво, а IntraWeb е class frameworks, разположен така, че да осигурява на разработчика абстрактен интерфейс от високо ниво. Разбира се и тук, този интерфейс може да бъде заобикалян, ако някаква допълнителна функционалност го изисква. IntraWeb прави почти ненужно директното обръщение към HTTP, HTML и Javascript, но и позволява достъп до ниското ниво, ако такъв е необходим. Така този подход осигурява както улеснение и бързина при програмирането, така и максимална гъвкавост. Нивата на Window/Linux "Class Frameworks" и IntraWeb Framework са дадени на фиг. 2.

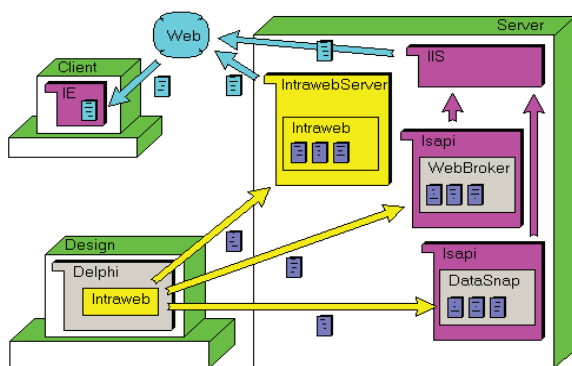


Фиг. 2 - Нива на Class Frameworks и IntraWeb Framework

На базата на мощен HTML модул за визуализация, IntraWeb позволява на разработчиците да проектират weblocation-и по същия начин, както биха разработвали нормални Windows-приложения. С помощта на drag-and-drop могат да се поставят компоненти-контроли в IntraWeb-форми (които могат да се разглеждат като комбинация между HTML-страници и Windows-форми), да създават събития и задават стойности на свойствата им. Кодът зад приложението може да е C#, Visual Basic, Delphi, C++ или Java, като в повечето случаи не се изискват никакви познания по HTML или JavaScript.

Посредством технологията IntraWeb е възможно създаването на два типа приложения:

- Stand Alone (SA) Server type – с интегриран собствен Web-сървър;
- ISAPI type – за работа с Internet Information Server (IIS) на Microsoft.



Фиг.3 - Дизайн, хостинг и достъп до IntraWeb-приложения

Процесът на IntraWeb дизайн, хостинг и клиентски достъп е показан на фиг.3.

Средата SpiderCloud е разработена като SA Server-приложение. При този подход функционирането на приложението е изключително опростено, като се избягват и проблемите по инсталиране и конфигуриране на web сървъри от рода на IIS или Apache. След компилиране в IDE Delphi се генерира IntraWebServer – изпълним обектен файл (SpiderCloud.exe), в който са интегрирани:

- HTTP Web Server (Indy-технология);
- Базов IntraWeb контролиращ код;
- Код на SpiderCloud – SPIDER online compiler.

За да започне да функционира средата е достатъчно стартирането на SpiderCloud.exe в произволен подходящ облачен Windows-сървър. Тъй като приложението съдържа в себе си HTTP web server, то по време на дизайна се настройва порта, на който ще се приемат заявки и ще се изпращат отговори. Нормално IntraWeb приложението се компилират за работа с порт 8888, но това лесно може да се промени. Портът по подразбиране за HTTP-комуникация е 80 и ако се използва ще бъдат елиминирани редица проблеми свързани с firewalls, proxy servers и др.

Клиентският достъп до така инсталираното облачно приложение става през произволен web browser, единствено посредством линк, включващ:

идентификация на облачния сървър : комуникационен порт

например:

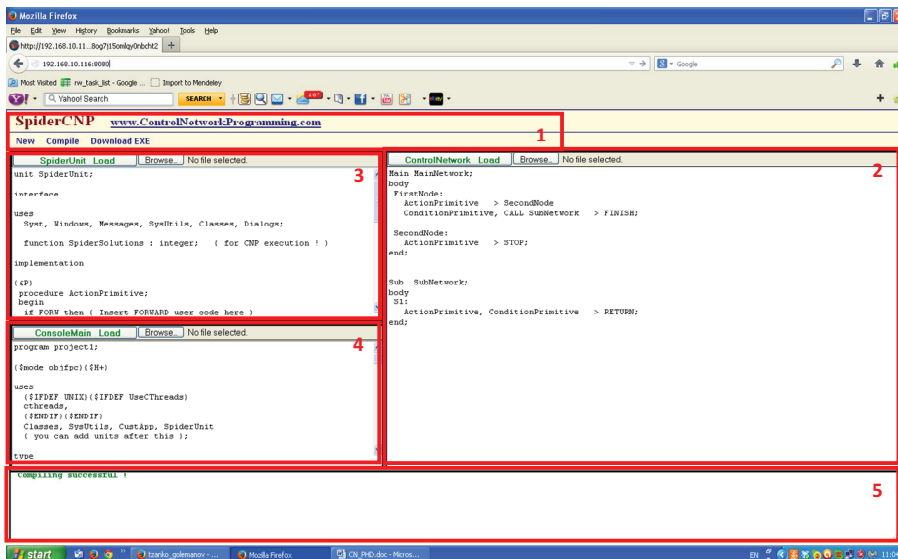
<http://ec2-54-241-120-135.us-west-1.compute.amazonaws.com:8080>

или

<http://54.241.120.135:8080>

Тук е необходимо е да се отбележи, че изобщо не е необходимо използването на многофункционална облачна платформа от вида на Amazon Elastic Compute Cloud (Amazon EC2) за разполагане на SpiderCloud – достатъчен е дори домашен компютър с разрешен за достъп отвън комуникационен порт.

Общият вид на потребителския интерфейс (зареден в Mozilla Firefox browser) е показан на фиг.4.



Фиг. 4 - Потребителски интерфейс на SpiderCloud

Обособени са пет функционални зони, включващи редакторски, контролни и комуникационни дейности:

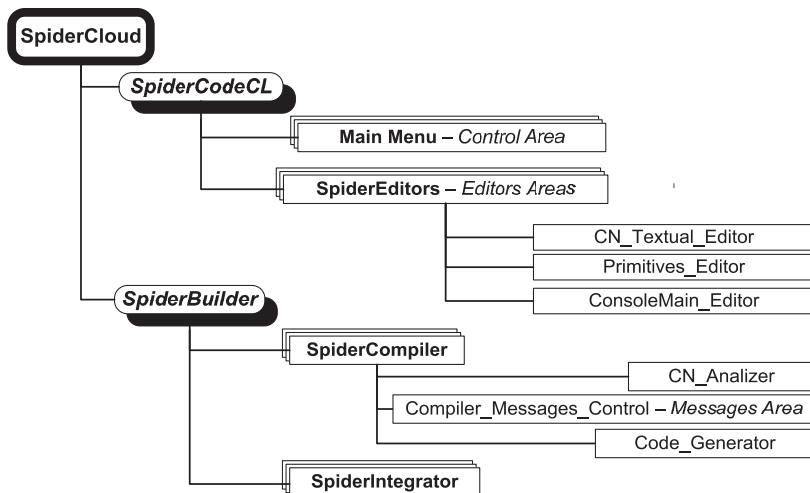
1. Control Area – управляваща област, включваща главното меню с опции:

- a. New - започване на нов (примерен, sample) проект с инициализация на редакторските области;
 - b. Compile – синтактична проверка и изграждане на текущия проект;
 - c. Download EXE – изтегляне (file download) на изградения обектен EXE-файл на текущия проект (след успешна компилация);
2. **ControlNetwork Editor Area** – зона на текстово зареждане и редактиране на Управляващата мрежа;
 3. **SpiderUnit Editor Area** – зона на текстово зареждане и редактиране на модула с примитиви и глобалните структури данни (типове и променливи);
 4. **ConsoleMain Editor Area** – зона на текстово зареждане и редактиране на главната програма на конзолния проект;
 5. **Messages Area** – зона за съобщения свързани с процеса на компилиране и изграждане на проекта.

Архитектура на инструменталната среда SpiderCloud

Модулната архитектура на инструменталната среда SpiderCloud е представена на фиг.5. Функционално системата се състои от две основни части (модула):

- **SpiderCodeCL** – представлява комплект инструменти за инициализация, upload/download и текстови редактори, осигуряващи необходимите средства както за задаване на Управляващата мрежа, така и за дефиниране на програмните примитиви, разделяемите структури данни и други програмни конструкции и модули;
- **SpiderBuilder** – използва се същия (от локалната инсталация [6]) независим модул (приложение) извършващ лексичен, синтактичен, семантичен анализи на SPIDER-програмата с индикация на локализираните грешки на различните нива. При успешно приключил анализ, модулът генерира междинната SPI-форма на CN-програмата, включваща управляващата мрежа с интегрирана логика на интерпретатора. Като следваща стъпка SpiderIntegrator автоматично прекомпилира всички файлове на проекта и изгражда финалния изпълним (EXE) файл.



Фиг. 5 - Обобщена йерархична структура на средата SpiderCloud

ЗАКЛЮЧЕНИЕ

Основно предимство на представеното облачно приложение е, че всички дейности по началното запознаване с езика SPIDER и технологията на програмиране чрез Управляващи мрежи (CNP) могат да се извършват през произволен web-browser. Този подход премахва необходимостта от инсталиране, конфигуриране и лицензиране както на средата на host-езика (Lazarus, Delphi), така и на настройването и интегрирането на локалната IDE SpiderCNP в нея.

Основните характеристики на SpiderCloud са:

- Стартиране като интернет-базирано приложение – SaaS услуга с осигурен многопотребителски конкурентен достъп;
- Опростен интерфейс, позволяващ зареждане и разработка на конзолни мултипарадигмени проекти;
- Разполагане на кода на отделните програмни парадигми (императивна и декларативна) в отделни редакторски текстови зони;
- Компилиране с указания за допуснатите грешки в различните части на проекта;
- Изграждане на проекта и възможност за download на генерирания изпълним обектен файл.

Предстои развитие на средата в следните направления:

- ◆ Потребителска регистрация и управление на историята на проектите;
- ◆ Добавяне на средства за настройка и тестване на разработваните проекти;
- ◆ Възможност за отдалечено изпълнение на компилираните проекти.

ЛИТЕРАТУРА

- [1] S. P. Mirashe and N. V. Kalyankar, "Cloud Computing," Communications of the ACM, vol. 51, no. 7, p. 9, 2010.
- [2] K. Kratchanov, T. Golemanov, and E. Golemanova, "Control Network Programming," in 6th IEEE/ACIS Conf. on Computer and Information Science (ICIS 2007), Melbourne, Australia, 2007, pp. 1012-1018.
- [3] B. Johnston, "Cloud computing is a trap, warns GNU founder Richard Stallman," The Guardian, 2008. [Online]. Available: <http://www.theguardian.com/technology/2008/sep/29/cloud.computing.richard.stallman>.
- [4] P. Mell and T. Grance, "The NIST Definition of Cloud Computing Recommendations of the National Institute of Standards and Technology," Nist Special Publication, vol. 145, no. 6, p. 7, 2011.
- [5] Morgan Stanley Research, "Cloud Computing Takes Off, MORGAN STANLEY BLUE PAPER," New York, USA, 2011.
- [6] Ц. Големанов, "SpiderCNP - интегрирана среда за визуално програмиране чрез управляващи мрежи," РУ & СУ 2012, 2012.
- [7] ATOZED Software, "IntraWeb - Web application development framework for Delphi and C++ Builder," IntraWeb - VCL for the Web, 2013. [Online]. Available: <http://www.atozed.com/IntraWeb/index.EN.aspx>.

За контакти:

Цанко Големанов, Катедра "Компютърни системи и технологии", Русенски университет "Ангел Кънчев", e-mail: TGolemanov@ecs.uni-ruse.bg

Костадин Крачанов, Department of Software Engineering, Yaşar University, Izmir, Turkey, e-mail: kostadin.kratchanov@yasar.edu.tr;

Емилия Големанова, Катедра "Компютърни системи и технологии", Русенски университет "Ангел Кънчев", e-mail: EGolemanova@ecs.uni-ruse.bg

Докладът е рецензиран.



РУСЕНСКИ УНИВЕРСИТЕТ „АНГЕЛ КЪНЧЕВ”
UNIVERSITY OF RUSE „ANGEL KANCHEV”

ДИПЛОМА

**Програмният комитет на
Научната конференция RU&SU'13
награждава с КРИСТАЛЕН ПРИЗ
“THE BEST PAPER”**

**ЦАНКО ГОЛЕМАНОВ, КОСТАДИН КРАЧАНОВ,
ЕМИЛИЯ ГОЛЕМАНОВА**

автори на доклада

**“SpiderCloud – Облачна реализация на среда за
Control Network Programming”**

DIPLOMA

**The Programme Committee of
the Scientific Conference RU&SU'13
awards the Crystal Prize "THE BEST PAPER"
to TZANKO GOLEMANOV, KOSTADIN
KRATCHANOV**

and EMILIA GOLEMANOVA

authors of the paper

**“SpiderCloud - A cloud implementation of
Control Network Programming environment”**

**РЕКТОР
RECTOR**

**Проф. д.т.н. Христо Белоев
Prof. DSc Hristo Beloev**

26.10.2013