



```
function RecoverDataAssistance($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD) {
    $objDataAssistanceBD = $this->startDataAssistanceBD($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD);
    $objDataAssistanceBD->Recover($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD);
    return $objDataAssistanceBD->GetDataAssistance();
}

function actualDataAssistance($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD) {
    $objDataAssistanceBD = $this->startDataAssistanceBD($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD);
    $objDataAssistanceBD->Recover($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD);
    return $objDataAssistanceBD->GetDataAssistance();
}

function insertDataAssistance($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD) {
    $objDataAssistanceBD = $this->startDataAssistanceBD($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD);
    $objDataAssistanceBD->Insert($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD);
    return $objDataAssistanceBD->GetDataAssistance();
}

function RecoverDataAssistanceBD($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD) {
    $objDataAssistanceBD = $this->startDataAssistanceBD($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD);
    $objDataAssistanceBD->Recover($nIdBenefit, $nSubsidized, $nSystemUser, $nDataAssistanceBD);
    return $objDataAssistanceBD->GetDataAssistance();
}
```

# Konferencja Optymalny znaczy najlepszy czyli, co nam dają nowe wersje oprogramowania?

Anna Olczak

## Nowości w CICS TS V4

18-19 listopada 2010  
Naruszewo, hotel Szkockie Ranczo

# CICS Deployment Assistant for z/OS

24 Sep 2010 – Witamy w klubie !!!!!



- Discover CICS regions and dependencies
- Check CICS topology and initialization options are as expected
- Work with JCL, spool files, and datasets
- Add a CICS region to an existing CICSplex
- Clone an existing CICS region
- Start and stop a CICS region

# CICS DEPLOYMENT ASSISTANT for z/OS



The screenshot displays the IBM CICS Explorer BETA interface. On the left, a project tree shows the structure for 'The new project MVS28'. The main area is divided into several panels:

- Overview (PLEX2.sysplex):** Shows MVS Images (MV29, MV2C, MV2D, MV28) and CICSplexes (GBPLEX1, T18PLEX2, T17PLEX, T18PLEX, CICSSEXP2, CICSSEXP3).
- CICS Regions:** Divided into Managed Regions (IYCWGIT0, IYCWFEFF1, IYCWEEEE1, IYCWFEFE1, IYCWFEFF2, IYCWWEED1, IYCWFEFD1) and UnManaged Regions (IYK3Z0JH, IYCWJGF1, IYK2ZHI1, IYCWJGE1, IYCWJGD1, IYK3Z0FF, IYCWFEFFS). It also shows CHAS (IYK2ZVF3, IYK2ZGV3, IYCWGIC0, IYCWEECM, IYCWFEFCM) and Web User Interfaces (IYCWFEFW2, IYK2ZGV4, IYK2ZVF1, IYCWEEW1, IYCWFEW1, IYCWEEW2, IYCWGIW0).
- Overview (GEMPLX01):** Shows Regions (IYCSGIA2, IYCSGIT0, IYCSGIT2, IYCWGIA0, IYCWGIT0), CHASes (IYCSGICM (MP), IYCSGIC0, IYCWGIC0), Web User Interfaces (IYCSGIW0, IYCWGIW0), and Groups (ALLCICS, AORS, GWGRPA, GWGRPA0, GWGRPA1, SILLY, SILLYA).
- Jobs:** A list of jobs including IYCSGIT1, IYCSGIT2, IYCSGIT0, IYCSGIT1, IYCSGIT2, IYCSGIT0, IYCSGIW0, IYCSGICM, IYCSGIT3, IYCSGIC0, IYCSGIT3, IYCSGIT3, IYCSGIT2, and IYCSGIT0.

The bottom status bar shows 'CNX01001 Connected user G8SHAL to host winmvs28.hursley.ibm.com on port 10203' and 'MVS28'.

Konferencja Optymalny znaczy najlepszy  
czyli, co nam dają nowe wersje oprogramowania?



# Dostęp z CICS Explorera



- CICS Transaction Server (CICS TS) for z/OS
- CICS Interdependency Analyzer for z/OS
- CICS Performance Analyzer for z/OS
- CICS Configuration Manager for z/OS
- CICS Transaction Gateway for z/OS, and for Multiplatforms
- IBM Application Performance Analyzer for z/OS
- IBM Debug Tool for z/OS
- IBM Fault Analyzer for z/OS
- IBM Rational® Developer for System z

## CICS products with EOS dates in 2009 through 2011

## EOS date

**CICS Performance Monitor for z/OS V1.2**

[03 Dec 2011](#)

**IBM Session Manager for z/OS V1.3**

[30 Apr 2010](#)

**SOAP for CICS feature**

[30 Sep 2009](#)

**CICS Transaction Server for z/OS V2.3**

[30 Sep 2009](#)

**CICS Interdependency Analyzer for z/OS V2.1**

[30 Apr 2009](#)

**CICS Performance Analyzer for z/OS V1.4**

[30 Apr 2009](#)

**CICS VSAM Recovery for z/OS V4.1**

[30 Apr 2009](#)

## CICS TRANSACTION SERVER for z/OS

GA DATE EOS DATE

[CICS Transaction Server for z/OS V2.2](#) 2.2.x [25 Jan 2002](#) [30 Apr 2008](#)

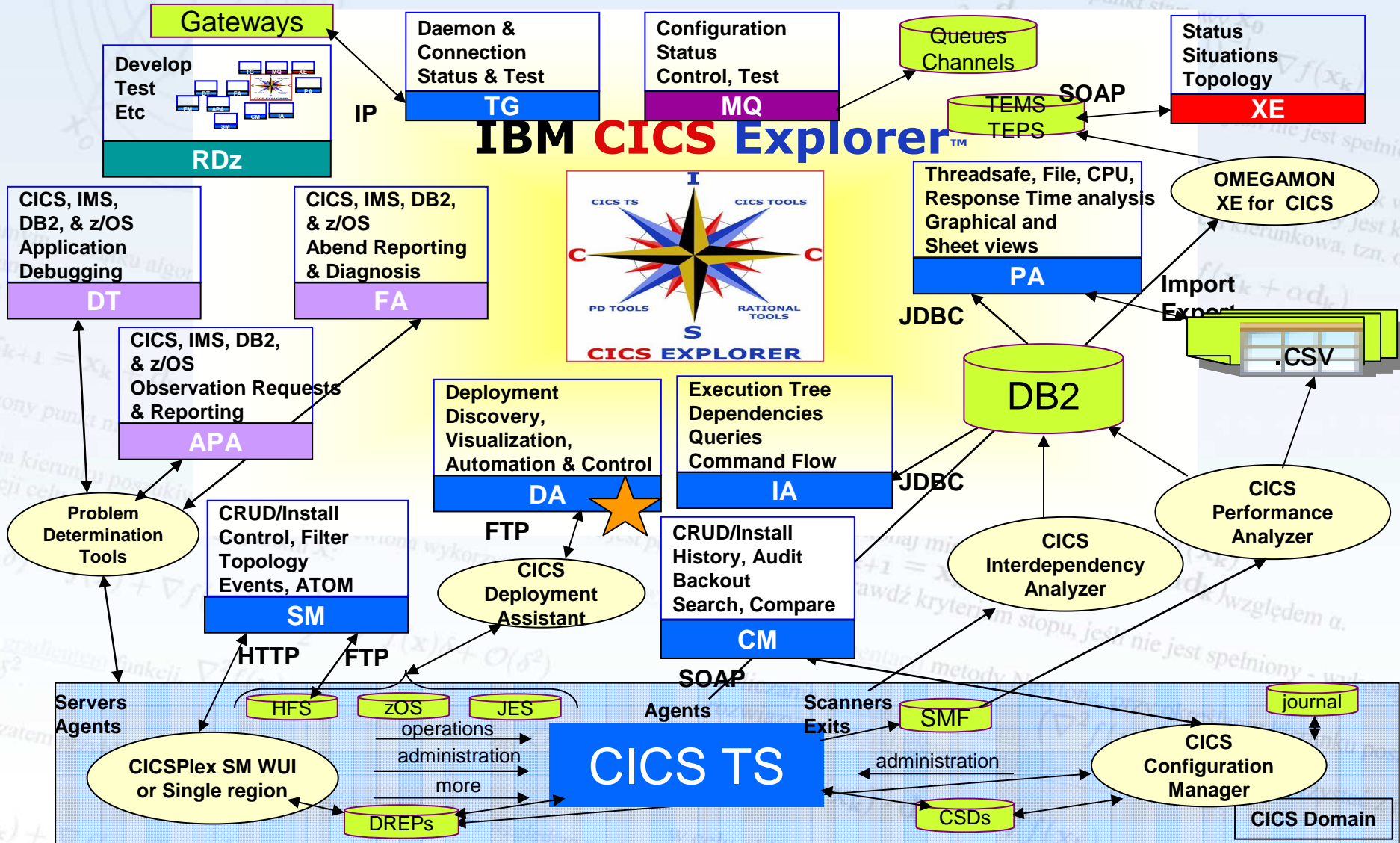
[CICS Transaction Server for z/OS V2.3](#) 2.3.x [19 Dec 2003](#) [30 Sep 2009](#)

[CICS Transaction Server for z/OS V3.1](#) 3.1.x [25 Mar 2005](#)

[CICS Transaction Server for z/OS V3.2](#) 3.2.x [29 Jun 2007](#)

[CICS Transaction Server for z/OS V4.1](#) 4.1.x [26 Jun 2009](#)

# IBM CICS Explorer™



SM	CICS Transaction Server
IA	CICS Interdependency Analyzer
PA	CICS Performance Analyzer
CM	CICS Configuration Manager
DA	CICS Deployment Assistant
TG	CICS Transaction Gateway

APA	Application Performance Analyzer
FA	Fault Analyzer
DT	Debug Tool
MQ	WebSphere MQ
XE	OMEGAMON XE for CICS
RDz	Rational Developer for System z

[ibm.com/cics/tools](http://ibm.com/cics/tools)

[ibm.com/cics/explorer](http://ibm.com/cics/explorer)

[ibm.com/cics/explorer/download](http://ibm.com/cics/explorer/download)





- Algoritm można zapisać:
1. Wybierz punkt startowy  $x_0$
  2.  $d_k = -(\nabla^2 f(x_k))^{-1} \cdot \nabla f(x_k)$
  3.  $x_{k+1} = x_k + d_k$
  4. Sprawdź kryterium stopu, jeśli nie jest spełnione

Algoritm w tym przypadku polega, analogicznie jak w przypadku punktu startowego. Dla danego punktu obliczany jest kierunek na jego podstawie minimalizacja kierunkowa, tzn. obliczamy:

$$f(x_k + \alpha_k d_k) = \min_{\alpha > 0} f(x_k + \alpha d_k)$$

Kolejny krok obliczany jest ze wzoru:

$$x_{k+1} = x_k + \alpha_k d_k$$

Algoritm można zapisać:

1. Wybierz punkt startowy  $x_0$
2.  $d_k = -(\nabla^2 f(x_k))^{-1} \cdot \nabla f(x_k)$
3. dokonaj minimalizacji  $f(x_k + \alpha d_k)$  względem  $\alpha$ .
4.  $x_{k+1} = x_k + \alpha_k d_k$
5. Sprawdź kryterium stopu, jeśli nie jest spełnione - wykonaj kolejny krok

Przy implementacji metody Newtona, przy określaniu kierunku poszukiwania, przy obliczaniu odwrotności hessianu  $(\nabla^2 f(x_k))^{-1}$ , warto skorzystać z numerycznego rozwiązywania układów równań liniowych:

$$\nabla^2 f(x_k) \cdot d_k = -\nabla f(x_k)$$

w celu obliczenia wartości wektora  $d_k$ .

# Dziękuję za Uwagę

W danym punkcie  $x_0 \in D$ . W punkcie tym obliczany jest kierunek poszukiwania  $d_k \in D$ . Punkt w następnym kroku obliczany jest ze wzoru:

$$x_{k+1} = x_k + d_k$$

Jeśli punkt nie spełni warunku stopu algorytm powtarza się. W metodzie Newtona wykorzystywane jest rozwinięcie Taylora funkcji celu względem danego punktu  $x$ :

$$f(x + \delta) = f(x) + \nabla f(x)^T \delta + \frac{1}{2} \delta^T \nabla^2 f(x) \delta + O(\delta^2)$$

gdzie  $\nabla^2 f(x)$  jest macierzą Hessera, zaś  $O(\delta^2)$  jest resztą rzędu drugiego. Wzrost funkcji  $f$  przy przesunięciu  $\delta$  można zatem przybliżyć przez aproksymację kwadratową  $F_2$  względem punktu  $x$ .

$$F_2(x, \delta) = f(x) + \nabla f(x)^T \delta + \frac{1}{2} \delta^T \nabla^2 f(x) \delta$$

aby znaleźć