

## SAP Support Package Implementations: Which Policy Should You Choose?

The development and use of software means the existence of bugs, and SAP® ERP is no different from any other software. Fortunately, SAP identifies these bugs through its support organization and provides a remedy through correction software updates called OSS notes (or simply ‘notes’).

Notes are grouped into support packages and support package stacks. For a detailed description of notes, support packages, and support package stacks see Appendix A. A company’s correction implementation policy defines at what level (single note, support package, or support package stack) corrections are implemented.

A recent survey<sup>1</sup> shows that the most dominant correction implementation policy among SAP customers involves implementation of complete support package stacks. 43% of the survey respondents are employing such a policy. Other respondents select only the required support packages (25%), apply only notes (17%), or implement corrections as part of an enhancement package (15%). See Figure 1:

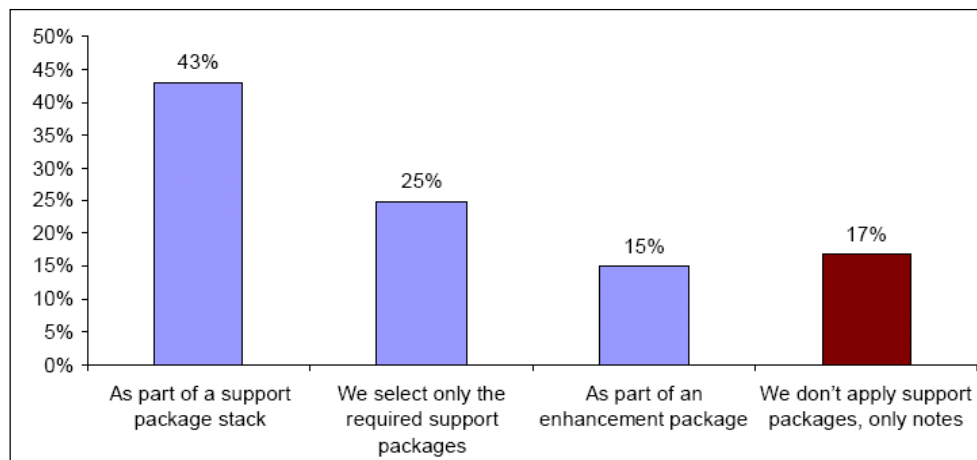


Figure 1 – Correction Implementation Policies

### Impact Analysis: The Key Correction Challenge

The actual implementation of support packages (and even support package stacks) is usually not a labor-intensive task. In fact, the technical aspects of implementing a support package stack should not take more than a day. But here’s the rub:

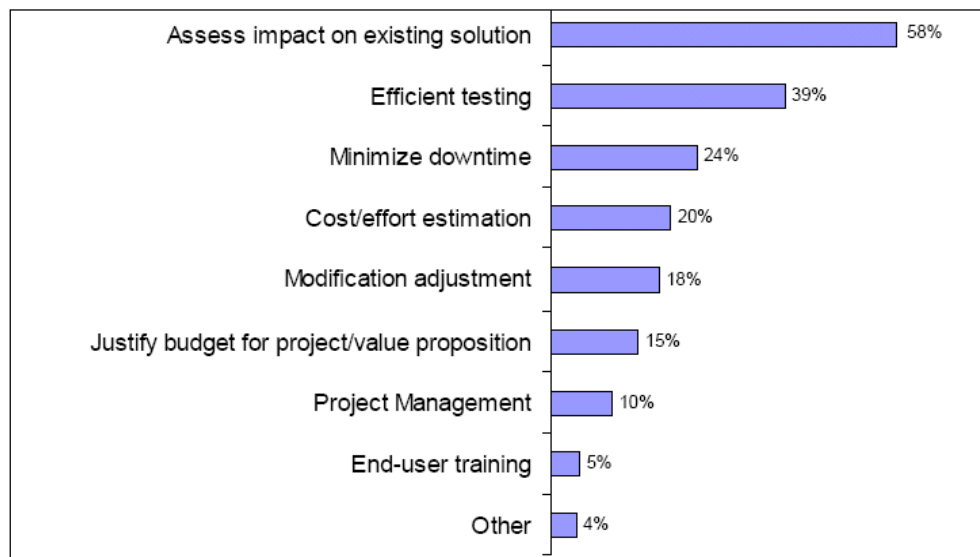
A single support package contains, on average, about 400 notes and a support package stack contains around 8,000 notes.

<sup>1</sup> 2009 SAP Support Costs Survey, <http://go.panayainc.com/SupportSurvey.html>

Some implementations span across several support package stacks and therefore translate to the implementation of tens of thousands of code corrections at once. For example, Support Package Stack 15 (the latest support package stack released by SAP to date) contains 8368 notes in 36 support packages.

When applying such a massive change to your code, the main challenge consists of assessing the impact of that change on the behavior of the system. By definition, notes are supposed to only correct bugs, but in reality they potentially create new problems by changing standard functionality, introducing new bugs, or, most commonly, causing custom code to fail.

For this reason, when asked what are the main challenges an organization faces when implementing support packages, 58% of survey respondents said the assessment of the impact on their existing solution. See Figure 2:



**Figure 2 - Support Package Challenges**

Consequently, support package implementation projects include extensive testing that is aimed at verifying the correct operation of the pre-implementation functionality and identifying any new problems introduced into the system. Not surprisingly, efficient testing was also selected as one of the key challenges in a support package implementation project by nearly 39% of survey respondents.

In regard to implementation timelines, the survey shows that support package implementations require an average of 73 person days, with the majority of the effort (42 days) associated with testing. See Figure 3:

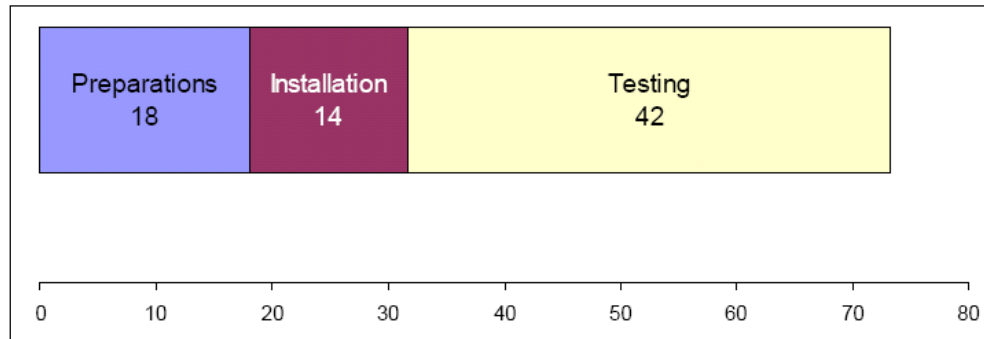


Figure 3 - Support Package Implementation Effort

## Correction Implementation Policies: Pros and Cons

The need on one hand, to correct bugs, and on the other hand, to handle impact analysis, has stemmed two different approaches to correction implementations in an SAP ERP system:

1. **Re-active Policy** – Notes are implemented as a reaction to bugs identified by the organization. Support packages and support package stacks are not implemented.
2. **Pro-active Policy** – Support packages (or support package stacks) are implemented on a regular basis to avoid problems before they impact the organization.

As expected, there are pros and cons to each of these approaches:

### Re-active Policy

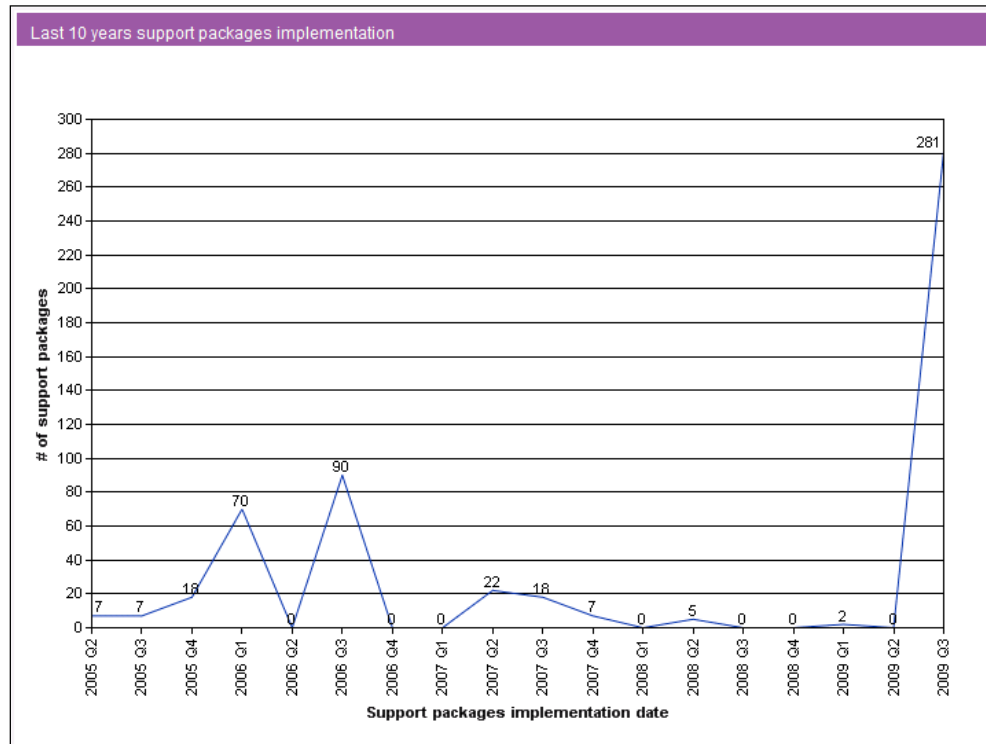
#### Pros

- Implementing individual notes is a non-disruptive event that is relatively easy to analyze and assess its impact. Therefore, implementations using this policy can be carried out in an ongoing manner without demanding extensive testing effort and with minimal risk of unexpected impacts.

#### Cons

- When support packages are not implemented, the SAP code becomes gradually outdated. This has two implications:
  - An increased likelihood of encountering errors, performance issues, and even security issues.
  - The implementation of an individual note often requires previous notes to be implemented first. These pre-requisite notes may, in turn, have pre-requisite notes of their own. The result over time is that instead of implementing individual notes, SAP support organizations find themselves having to implement long threads of notes. This puts pressure on SLA time frames and complicates the notes' impact analysis.

To demonstrate this policy, one SAP customer chose to implement very few support packages during the past three years and instead implemented 278 individual notes in his ERP system (see Figure 5). Upgrading this system to the latest support package stack will require this customer to implement 281 support packages at once. See Figure 4:



**Figure 4 - # of support packages implemented over time using a re-active implementation policy**

Note number	Description	Implemented with SNOTE	Valid For Target	Change r
Count: 278				
1059421	CUTRC: Inconsistencies	Yes	No	T30K9A2:
1066662	Incorr. depreciation on FA & subsequent acquisition for LVA	Yes	No	T30K9A1:
1154195	Revert back corrections done on initial download	Yes	No	T30K9A1:
1035632	Asset history sheet during mid-year complete retirement	Yes	No	T30K9A1:
1100403	FTWD - Data checksum does not perform a regular commit.	Yes	No	T30K9A1:
1226171	Identifying jobs with deleted printer	Yes	Yes	T30K9A2:
1141312	View file is created without header	Yes	No	T30K9A1:
850063	External tax system: store modified tax base amount	Yes	No	T30K9A1:
1092942	FTWE/FTWE1: Negative sign is not displayed correctly	Yes	No	T30K9A1:
1075200	Purchasing document activation checks pro forma invoice	Yes	No	T30K9A1:
1078245	Dump NODE_NOT_FOUND when displaying document flow	Yes	No	T30K9A0:
1033419	FB03/FBL1N: Attachment is not saved in line item	Yes	No	T30K9A0:
1008166	Setting the goods movement status for inbound delivery	Yes	No	T30K9A1:
1123828	ME21N, ME22N: Delivery date copied from first item line	Yes	No	T30K9A1:
1086081	Classification: Unnecessary batch lock	Yes	No	T30K9A1:
1017089		Yes	Unknown	T30K9A0:
1060049	Index VRKPA when archiving invoice lists	Yes	No	T30K9A1:
1113847	Buffering - Characteristic value assignment screen III	Yes	No	T30K9A1:
687329	Third-party with CC: Incorrect values in KW00 and PCVP	Yes	No	T30K9A0:

Figure 5 - Notes implemented using an infrequent support package implementation policy

## Pro-active Policy

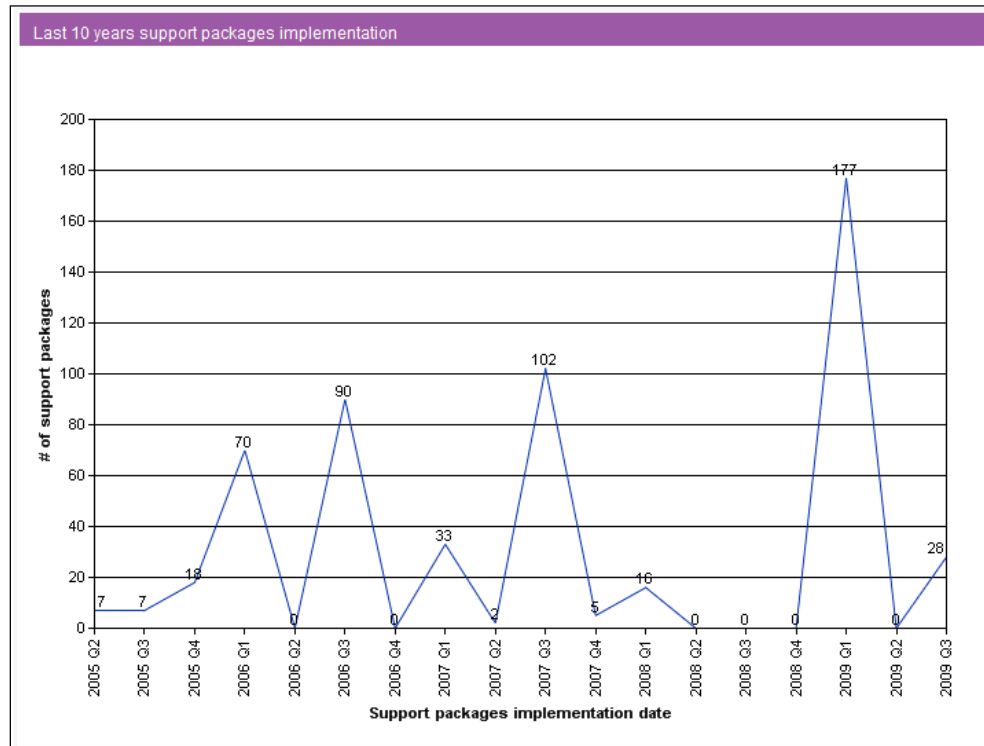
### Pros

- Keeping your code up-to-date is always a good practice and probably the best way to avoid errors on an ongoing basis.

### Cons

- Since implementing support packages requires extensive testing and also some level of code freeze throughout the implementation project, it inevitably presents a disruption to the IT project plan. Organizations find it difficult to frequently stop everything they are doing and shift their focus to support package implementations.

To demonstrate this policy, one SAP customer implements support package stacks regularly (on average, every 6 to 12 months) and has had to implement only 44 individual notes. Thanks to his keeping his code up-to-date, upgrading to the latest support package stack will include the implementation of only 28 support packages. See Figure 6 and Figure 7:



**Figure 6 - # of support packages implemented over time using a frequent support package implementation policy**

Note number	Description	Implemented with SNOTE	Valid For Target	Change request
Count: 44				
1036181	Unauthorized syslog entry of type H0/1	Yes	No	DV2K9I
851545	Distributing manually entered tax amount to tax line items	Yes	No	DV2K9I
1016171	Unauthorized syslog entry of type H0/1	Yes	No	DV2K9I
1025130	Success message missing in the BAPI for purchase orders	Yes	No	DV2K9I
1040693	Unauthorized syslog entry of type H0/1	Yes	No	DV2K9I
1071154	EDI: Decimal place error in E1EDP05/E1EDK05-BETRQ	Yes	No	DV2K9I
1174114		Yes	Unknown	DV2K9I
997941	Avoid recalculation of time and date of splits	Yes	Yes	DV2K9I
1017922	OBJECTS_MOVE_NOT_SUPPORTED in SBAL_INTERNAL_UCOMM_ANALYSE	Yes	No	DV2K9I
1014555	FEBA_LOCKBOX Posting Log - RFEBBU00 (spool).	Yes	Yes	DV2K9I
1087879	Performance issues while archiving using SD_VBAK & SD_VBRK	Yes	No	DV2K9I
1020403	Lockbox Posting Logs require header for overflow pages.	Yes	No	DV2K9I
1031897	Dump when split detail changes	Yes	No	DV2K9I
1079390	PRICING_COMPLETE may determine an incorrect net value	Yes	No	DV2K9I
1001282	ME33K, ME2V (ALV): Different problems in reporting	Yes	No	DV2K9I
1057877	ME21N External Details not filled automatically	Yes	No	DV2K9I
1001645	SAP AS: Performance when setting up an index	Yes	No	DV2K9I
833303	FB01, FB80 dump in case of cross-company-code postings	Yes	No	DV2K9I

« < Page 1 of 3 > »

**Figure 7 - Notes implemented using a frequent support package implementation policy**

## Hybrid Policies

In an attempt to seize the better of the two policies described above, some SAP customers adopt hybrid policies such as:

- Implementation of support packages when SAP creates a strong motivation to do so; e.g., when the support package is a pre-requisite for an enhancement package implementation.
- Update regularly only a small set of software components, e.g., keeping only the HR component up-to-date, which is often mandatory for regulatory compliance, while keeping the rest of the system unchanged.

## Recommendations

Although there is no absolute right or wrong correction implementation policy, after analyzing the behavior and consequences of hundreds of SAP customer policies, we can suggest the following guidelines:

- **Don't lag too far behind** – Waiting longer than two years between correction implementations will result in an effort that is almost comparable to a full-fledged release upgrade (which requires a long code freeze period), an organization-wide challenge to find the right time to implement the support package stacks, a riskier project, and an increased post go-live risk.
- **Keep track of 'special notes'** – Although most notes are classified as 'Program Errors' and are aimed at correcting bugs, some notes are classified differently (e.g., 'Advanced Correction' or 'Workaround for missing functionality') and provide special functionality. Our SAP customer analysis shows that the latter tend to be more frequently and severely impacted by support package implementations and upgrades. It is a best practice to keep track of any such notes implemented in your system and verify that the functionality they affect still works after the upgrade.
- **Use automation tools for impact analysis** – Given that impact analysis is the main challenge organizations face when implementing corrections, we recommend investing in automation tools to remove most of the complexity, risk, and effort otherwise inherent to these projects.

### Get Your Free Support Package Automation Trial Today

Click here for more information about SAP support package automation:

<http://www.panayainc.com/SAP-support-packs-SAP-support-packages-Panaya.html>

Click here to sign up for your free support package automation trial:

<http://www.panayainc.com/Request-a-Trial.html>

## Appendix A – Notes, Support Packages and Support Package Stacks

**Notes** include code corrections and/or data-dictionary changes to SAP standard objects. These changes are called correction instructions.

Each note is aimed at solving a specific problem across all releases in which this problem exists. A single note may include multiple sets of correction instructions - one per each release containing the problem. For example, a note may include two instruction sets - one for correcting the problem on a 4.6c system and the other for correcting it on a 4.7 system. Ultimately, the correction is incorporated into the standard code of a given release that makes the note obsolete for that release and future releases.

Notes are released regularly and frequently – between 2,000 and 3,000 notes are released every month just for SAP ERP. Since individual implementation of notes is labor-intensive, SAP groups notes into **support packages**, which customers can implement in a single run.

Support packages target a specific application component and are released sequentially. For example, the EA-HR ERP component provides HR-related functionality. Every month a new support pack is released for this component.

There are dependencies between ERP components that create subsequent dependencies between support packages. To tackle this reality, SAP introduced the concept of **support package stack** that aligns support packages across all ERP components. Support package stacks are released (roughly) on a quarterly basis.

### About Panaya

Panaya's Software-as-a-Service solutions enable companies that use SAP to save up to 50% of their application lifecycle costs and minimize the risks associated with system changes. Utilizing cloud-based simulation to analyze the impact of pending changes, Panaya automatically pinpoints which custom programs will break as a result of an upgrade or support package implementation. Panaya provides a complete solution for managing these changes, explaining how to fix the anticipated issues, suggesting the most efficient test plan, and calculating required project budget and resources.