

▪ پیکر بندی و ارتقاء

Enterprise Software جنبه های مختلف شرکت را به هم متصل می کند به طریقی که هر Transaction به صورت مجموعه ای از اطلاعات برای شرکت در دسترس باشد. نرم افزار ، سیستمی مطابق با business است . برای راحتی در برنامه نویسی و استفاده ، نرم افزار به صورت تعدادی ماژول که روی قسمت خاصی از شرکت تمرکز می کند خواهد بود . با وجود اختلافاتی که بین provider ها هست ، هفت تا ده ماژول تحت عناوین :

مالی و حسابداری ، زمان بندی مدید ، منابع انسانی ، مدیریت انبار و ... وجود دارد . هر کدام از اینها باید مناسب opration خاص و نیازهای کسب و کار باشد. به هر حال در بعضی موارد ارتقاء به یک پکیج نرم افزاری به منظور پشتیبانی و حمایت از کار کرد های مهم و بحرانی کسب و کار ضروری است . هر ماژول باید یک تیم طراحی ES داشته باشد که کارکرد های شرکت را در آن ناحیه منعکس کند . این گروه با تیم پروژه ERP فرق دارد .

در یک پیاده سازی ترکیبی ERP/ES یک چالش ، نگه داشتن تیم طراحی ES در کنار تیم در کنار تیم ERP است . یکی از بهترین راهها برای انجام این کار حفظ مراتبی از عضویت مشترک است . یک یا چند عضو از تیم طراحی ES می توانند

به سازمان ERP مربوطه تخصیص داده شوند و یا بالعکس یک تفاوت عمده بین تیم ERP و ES این است که تیم ERP روی افراد و اطلاعات تمرکز می کند در حالی که تمرکز تیم ES روی نرم افزار و سخت افزار است .

به هر حال هر دو درگیر طراحی دوباره فرآیند های کسب و کار هستند .انتخاب تیم طراحی حساس و ضروری است . برای ضویت در چنین تیمی هر فرد باید قادر باشد که برای سازمان تصمیم گیری کند و همچنین بتواند راجع به مسائلی که برای سازمان مهم است و باعث پیشرفت آن می شود گفتگو کند . اعضای تیم باید با واحدهای داخلی در تماس بوده و پیشنهادات و نظرات خود را به آنها منتقل کنند آنها نقش نماینده واحدها را دارند و می توانند از جانب آنها تصمیم گیری کنند . سوالی که پیش می آید این است که این تیم بزرگ چگونه باید باشد ؟ تیم کوچکتر بهتر است و می تواند با حداکثر ۲۰ نفر دایر شود . ولی رهبری یک تیم بزرگ دشوارتر است . در یک تیم کوچک اگر اهداف روشن نشده باشد و رهبر تیم مشخص نباشد ممکن است اختلافاتی به وجود بیاید . تیم بعضی از قسمت های نرم افزاری رهبرانی از گروه IT دارند . در بعضی موارد استخدام مدیر از گروه های دیگر هم می تواند موثر واقع شود . مثلاً کسی از قسمت خرید

و فروش می تواند برای مدیریت تیم طراحی مفید و موثر باشد . در پیاده سازی ERP/ES برای staffing شما با دو انتخاب روبرو هستید .

۱- ترکیب تیم طراحی ES و گروه پروژه ERP و به حداقل رسانده اعضای اصلی

۲- تمرکز روی پروژه ES و به دنبال آن پروژه ERP یک حالت این است که ES را بدون ERP بخواهیم نصب کنیم و همچنین توانایی staffing تیم طراحی

را نداریم در این صورت بهترین انتخاب این تصمیم گیری در مورد تعداد تیم ها و انجام یک پروژه چند فازه است . در این حالت باید ۲ یا ۳ واحد اصلی را انتخاب

کرده و تیم را تنها برای آنها برپا کنید . مشاوران نرم افزار می توانند در این پروسه به شما کمک کنند . اما آنها به

سادگی نمی توانند افراد با هوش و مطلع از اوضاع شرکت را جایگزین کنند . وقت زیادی بگیرد . زیرا مشاوران برای درک کسب و کار باید با افراد درون شرکت

مصاحبه و گفتگو کنند . بهترین راه این است که از تعداد کمی مشاور استفاده کنیم که بتوانند به تصمیم گیری های تیم کمک کنند . بدون این که تخصص لازم را در

زمینه عملکرد شما داشته باشد .

■ Installation

توصیه هایی راجع به نصب نرم افزار :

- انعطاف پذیری ، اگر انعطاف پذیری وجود نداشته باشد

ممکن است نصب دقیق آن چه که تیم طراحی تعیین کرده

کار سختی باشد . کار نکند زیرا تلاش های تیم طراحی ES

ممکن است با هم سازگار نباشد . این ناسازگاری می تواند

بین تیم طراحی ES وجود داشته باشد یا با تیم پروژه

.ERP

- آزمایش از این طریق می توان فهمید که آیا طرح تیم

درست کار می کند یا نه اما با این کار همه چیز مورد تایید

واقع نمی شود در این مرحله نرم افزار می تواند به راحتی

تغییر داده شود.

- سنجش حتی اگر در مرحله آزمایش اطمینان پیدا کرده اید،

نباید ES را یکباره در شرکت به راه بیندازید. بهترین راه

این است که قسمتی از کسب و کار را انتخاب کنیم چون

خطرات کار این کار از نصب یکباره کمتر است.

بعضی از شرکت ها تنها یک ماژول را در شرکت نصب می کنند که اینکار هم خطرات زیادی دارد. برای مثال نصب ماژول انبار و یا توزیع به تنهایی ممکن است به نظر کم خطر بیاید. این ماژور می تواند توسط تیم طراحی پشتیبانی شود. مشکل این است که خطاهایی که در موقع set کردن تغییرات به وجود میآید میتواند کارهای حمل و نقل شرکت را متوقف کند. خطرات Pilottest با چند عامل می تواند کاهش داده شود.

اول اینکه در قسمتی از کسب و کار است و دوم اینکه شما تمام منابع را در برابر ناحیه آزمایش در دسترس قرار می دهید.

واحد Pilottest در اتاق کنفرانس pilot به بحث بنشینند. تیم های طراحی در طول فرایند، از اتاق کنفرانس pilot تا rollout باید بدون تغییر باقی بمانند و در این مرحله نصب لازم نیست عمیقانه درگیر می شوند اما برای مشاوره لازم است که در دسترس باشند. کسی بهتر از آنها عملکرد ماژول ها را نمی داند. همچنین وجود اعضای تیم طراحی برای موفقیت پروژه نهایی مهم است و نه فقط در مرحله طراحی.

▪ پشتیبانی

یک اشتباه بزرگ در نصب نرم افزار اصلی این است که وقتی یکبار که نرم افزار اجرا شد پروژه را تمام شده در نظر بگیریم. اما نکته ای که وجود دارد این است که به هنگام تغییر شرکت نرم افزار هم باید به تبع آن تغییر داده شود. همان طور که گفته شد گروههای دپارتمان IT مدیران اطلاعاتی هستند نه نویسندگان نرم افزار. حال آنها چگونه این کار را انجام می دهند؟ تغییر بزرگ از سیستم های چند پاره قدیمی تا شرکت های وسیع جدید این است که نرم افزار نقش یک سیستم مرکزی برای شرکت را دارد. به علاوه تغییرات در bussiness strategy باید منعکس و جمع آوری شود. provider ها دائماً باید ورژن های جدیدی از نرم افزار را منتشر کنند که بعضی از آنها ممکن است برای bussiness شما مفید واقع شوند. نکته مهمی در رابطه با منابع تکنولوژی اطلاعات وجود دارد. در گذشته بسیاری از واحدهای کسب و کار کنترل افراد IT خود را به عهده داشتند. داده نگه داشتن سیستم متمرکز شده ضروری بود. به هر حال ES ها یک database مرکزی دارند از این رو به یک سیستم قابل اطمینان و شبکه های واضح و روشن نیاز است. برای جلوگیری از تکه تکه شدن مجموعه نرم افزاری باید منابع IT را به صورت مرکزی کنترل کنیم. اگر واحدهای محلی کنترل منابع IT خود را برعهده داشته باشند احتمال وقوع مغایرت وجود

دارد. واحدهای محلی به منابع IT نیاز دارند برای اطمینان پیدا کردن از این که آنها به طور مؤثر از سیستم های اطلاعاتی استفاده می کنند و برخورد با تغییرات نیازمندی های در کسب و کار. در هر صورت افراد IT باید یک گروه IT مرکزی به عنوان organizational داشته باشند.

▪ BOLT-ON Software

شرکت ها معمولاً bolt-on را به سیستم اصلی اضافه کنند. برای اجرای function های خاص زیرا ES موجود یا هستیم باقی مانده آنها را به خوبی اجرا نمی کنند یا اصلاً اجرا نمی کنند. بسیاری از پکیجهای نرم افزاری Bolt-on به عنوان best of bread مطرح شده اند. زیرا برای ماژول های همتای خود در Enterprise system ممتاز به نظر می رسند.

Bolt-on پایین آمدن توانایی ES در یکپارچه سازی اطلاعات و فرایندها، احتیاج به فایل های اضافی لینک نشده به data base مرکزی، تلاش برای یکپارچه سازی Bolt-on و نگهداری و ابقاء Enterprise system و Bolt-on بعد از تغییرات را در بر دارد. این تاثیرات منفی نا چیز نیستند. از این Bolt-on باید به طور صحیح و فقط در مواقع لزوم استفاده شود. اغلب Bolt-on هایی که در محیط

ERP دیده می شوند ۳ نوع هستند :

• Resource Planning Enablers

• Front-end / back-end : این application هایی هستند که روی

Front-end فرآیند برنامه ریزی منابع متمرکز می شوند (پیش بینی

فروش، فروش و برنامه ریزی عملیات و موجودی (vender- managed)

یا روی مانند پکیج های زمان بندی محدود.

• Supply chain optimization / advanced planning system

که برای یک میزان سازی بهتر جزئیات تقاضا شده توسط زمان بندی

مدیر و برنامه ریزی نیازمندیهای مواد، تلاش می کند. به واسطه منطق

شبیه سازی قوه آنها می توانند پیشنهادهای بهتری برای تقاضاهای

مدیران، طراحان، برنامه ریزان، مشتریان و استفاده از منابع داشته

باشند.

▪ مدیریت درخواست ها برای تغییرات.

با انتخاب ERP / ES و یا ERP به تنهایی شما درخواست هایی برای

تغییرات نرم افزار دارید. سالهاست که میلیون ها دلار صرف مشاوره، افراد

متخصص در زمینه نرم افزار و برنامه نویسان برای اصلاح پکیج های نرم

افزاری می شود. امروزه که به سمت فرآیند های کسب و کار (ERP) و

نرم افزار های رایج (ES و نرم افزارهای پشتیبانی زنجیره تولید) پیش می رویم، کنترل تغییرات را به دست داشتن یکی از چالش هاست.

اگر ERP به درستی آموزش داده شود تقاضا برای تغییرات به حداقل می رسد. و کاربران می توانند در غالب چارچوب ERP مشکلات خود را حل کنند. کاربران خواهند فهمید که چرا نرم افزارهای برای پشتیبانی نیازهای ERP شکل گرفته اند. در اینجا مدیریت نقش بسیار مهمی را ایفا می کند. افراد کلیدی، اعضای مدیریت و تیم پروژه به موارد زیر احتیاج دارند :

(۱) اصول

- مخالفت در برابر تغییرات مجزا

مدیریت باید از تغییرات نرم افزارهایی که برای انجام کسب و کار و یا

پیاده سازی ERP ضروری نیستند جلوگیری کند. تغییراتی که در طول

پیاده سازی پروژه به وجود می آید پروژه را به تأخیر انداخت و تغییراتی

که بعد از پیاده سازی پروژه ایجاد می شود کاربران را سردرگم می کند.

- همیشه فرآیند تغییرات شناخته شده را دنبال کنید. راه جلوگیری از نقض

اصل اول این است که یک فرآیند شناخته شده برای تغییرات داشته باشیم.

مدیریت باید فرآیند تغییرات را به روشنی با تمرکز بر موارد زیر مقدور کند : اینکه چه کسی خواستار تغییرات است، مطالب کلیدی مطرح شده چیست و چه کسی با تغییرات موافقت می کند.

(۲) طرز عمل

- دپارتمان IT ، اصلاحات و تغییرات را آماده می کند که شامل تغییرات بوجود آمده داخلی و انجام شده توسط فروشنده نرم افزار می باشد.

- تقاضا برای تغییرات برای ارزیابی میزان کاری که برای انجام آنها لازم

است به دپارتمان IT ارائه داده می شود. باید مرزی بین حداقل و حداکثر

تغییرات پروژه وجود داشته باشد و درخواست ها طبقه بندی شوند. IT

همچنین توضیحاتی درباره مسائل فنی تغییرات اضافه میکند ولی درباره

صحت کسب و کار توضیحی نمی دهد.

- تقاضاها سپس به تیم پروژه می رود. اگر تقاضا برای تغییرات کمی

باشد تیم پروژه تصمیم می گیرد که آنها را بپذیرد و یا به فاز III به تعویق

بیندازد.

- اگر تقاضا برای حداکثر تغییرات باشد تیم پروژه آن را مرور کرده و

تصمیم می گیرد موضوعات کلیدی موارد زیر است :

آیا این تغییرات برای انجام کسب و کار و یا درست کار کردن ERP ضروری است آیا آن عملکرد مورد تقاضا به کامپیوتر احتیاج دارد و یا به طور دستی

هم قابل اجرا است؟

اگر جوابها معلوم کند که این تغییرات برای کسب و کار مهم است و به کامپیوتر احتیاج دارد باید در اسرع وقت انجام شود اگر نه به فاز III به تعویق انداخته می شود.

• پذیرش تغییرات ممکن است توسط تیم پروژه رد شود. در این حالت

فرآیند تغییرات باید به کمیته رهبری فرصت داده شود تا در این مورد

تصمیم گیری شود. ایجاد تغییرات در ورودی و خروجی به مراتب

ساده تر از ایجاد و تغییرات در منطق سیستم است. هرگونه تغییر در

منطق سیستم باید به طور وسیع طرح شود و به حداقل برسد.

بسیاری از تغییرات در نرم افزار نشان دهنده این است که شما

نرم افزار غلطی دارید. ES ها به منظور تعدیل تغییرات ساخته شده

اند که جریان داده ها را کنترل می کنند.