The mysteries of retrospective computer system validation

What is a retrospective computer system validation?
How can it performed and documented?
What is the expectation and result of such a validation?
Dos and Don'ts ?

We are very often asked about the retrospective validation approach. So this short article describes some basics and careful consideration about this special topic and related concepts and interpretations, with making no claim to be complete.

In general the pure theory of computer system validation is based on a prospective approach, which enables a foresighted, projectable, changeable, and rectifiable method of system and vendor selection, development, implementation, testing, and operation along the entire lifecycle of the computerised system. In addition responsible personnel (e.g. QP, Quality Unit etc.) can pre-approve the release and usage of applications and systems before they will be used as part of GXP regulated activities (e.g. replacing manual operations).

Senior management has the ultimate responsibility to ensure an effective quality management system is in place and Quality Control should ensure that the appropriate validations are done. The quality system should be based on the related predicate rules, and these require the prospective validation of computerized systems or applications.

Therefore it is pretty much clear that it is impossible to implement and release an IT system first and later to validate it retrospectively during operational usage. This would show that the company is not following its own rules or is doing business out of compliance.

In general the agency's expectation is that systems are validated prospectively. But there are also good or at least suboptimal reasons to execute a retrospective validation.

First of all there are also different other terms, like e.g. re-validation, which we define here as a recurring validation of an already validated system, in order to re-validate the system because of e.g. major changes to the software or platform. We use the term retrospective validation only for systems that never have been validated and will be, have been or are currently in operations in non-GXP or GXP-related activities or processes.

What was/is the trigger for a retrospective validation? Two principal situations

A good reason for a retrospective validation is for example when a pharmaceutical company acquires a chemical or food company, or is planning to use other internal non-GXP business units of the company, or any similar case in order to produce and extend their GXP production on such a basis. Then a retrospective validation can be proactively planned – "prospectively" - , unless no GXP activities have been performed previously. At the time of starting the retrospective validation there is no risk or impact on patient safety or product quality. Typically a kind of a Quality Integration Program or similar will be initiated, in accordance to a change control process (record), where such a retrospective validation can be executed. It is also acceptable that historically data can be used to provide evidence in this context, as long as the processes and conditions are comparable or similar.

Such a history or experience report might show for example that the system was used in the past 5 years and produced 212 chemical batches, and no deviations or OOS were caused by the system (on functional level) during this time period. This sounds quite good for the first moment, but unfortunately there is also a small "but" with it. If the system was never run under a controlled and documented change process (release mgt.), or even worse was changed last week, this result can not be used for the planned system operations. It might be useful to check the invoices of the supplier indicating changes over the last years, if this is useful.

Also this is only showing the functional aspects of a system – still the procedural requirements like incident, change, access & security, training, data integrity etc. should be covered according the classical V-model. There might be decisions or discussion required, if in such a case a User Requirement Specification (URS) or a Supplier Audit is really mandatory, because the system is already in place (as-is). If it can be proofed that the Process Requirements (e.g. process maps) and appropriate Quality Agreements are in place, both steps might be skipped or reduced. In any case such a quality decision requires a risk-based assessment and written statement (rational), e.g. in the Validation Plan or Report. In addition it should be noticed that the retrospective validation of a system should go from hand in hand with the process validation or qualification and other quality assuring activities (e.g. basic GMP training, documentation practices, etc.).

The "ugly" situation for a retrospective validation

The above mentioned case is reasonable and plannable – risks are under control and can be proactively managed. Other cases requiring a retrospective validation might be:

- ➔ The GXP relevance of the system was detected too late
- ➔ The required validation of the system was forgotten, the system just went operational

This means that the system has already manufactured products, tested samples, created records etc. (GXP activities) which have been released to the market and/or delivered to the patients.

Another critical point is who and at which time this gap of a non-validated system (and/or process) is detected – first of all it is great that it was found, but what should be the next steps.

If an inspector or auditor is finding this gap, this is really uncomfortable and embarrassing – and if the gap is also existing since many years. At this point the entire "effective quality management system" can be questioned and challenged, which should be provided by the – competent - Senior Management. The response letter should contain a complete clarification and examination of such a finding /observation. Anyhow a complete or partially recall of all related batches might be required or prepared, if any evidence is not achievable and presentable. Any complaint record should be investigated in detail – don't bring forward the argument that there was no negative feedback from the market before taking such a serious action in details (documented assessment).

There are many different aspects in starting a retrospective validation. But one of the first questions should be if patient safety or product quality is affected (right now). On the other hand it should be analysed if the system does have any direct or indirect impact on safety and quality.

In general if a system is close to the production process or final product, this is definitely risky. If an electronic document management system for SOP management is not validated and is in operation since only 2 week, this might be seen as a poor result, but not highly critical and a roll-back to the former paper system might be possible in some hours/days. Or you run 15 similar HPLC systems from the same vendor and identical software, and 1 of them was not validated (for whatever reason), this might be not that critical as you find a non-validated ERP system managing the creation of batch numbers, orders, recipe, routing, warehouse management (FEFO), and electronic batch releases. So is the captain now talking about a yellow or red alert? In any case this should be investigated in detail and properly documented. Every case or variant of a retrospective validation is very special and requires a full risk mapping from the process, pharmaceutical product, software product, timing & schedules, delivery status, involved persons or vendors and much more. The difficulty is that the whole organisation (all roles at all levels) is plunged in at the deep end and it is logically not possible to plan quality and compliance or to arrange a risk-based avoidance strategy afterwards. In an assessment for a retrospective validation you may have a change of mood on an hourly basis, from finding extremely critical issues and then to give the all-clear in the next step. A very clear and transparent project plan should be used as a leading road-map through this process for all involved subject matter experts, where tasks, results, and decisions are controlled, communicated, and documented. And don't shoot the messenger(s): All information, expertise, comments and views are important and parts of a puzzle, where you act a little bit like an agent for the "Validation Scene Investigation".

How to solve the "ugly" situation? There's no point crying over spilt milk

First action in such a case is to initiate a deviation process (record) immediately. This must be a critical deviation (out of compliance) and clearly communicated – if it is reasonable at a later time the deviation can still be down-graded to a lower level. Any planned change to the system needs to be restricted – you need to look right now on a frozen system as a validation and investigation object (no moving target).

An initial risk assessment of the system should be quickly done, at least on a high level. So if you find out that there is no back-up procedure or system access control, you should remediate this first before starting to write a URS document according the V-model approach. The sequential order of a prospective validation is not mandatory or efficient for a retrospective approach.

It is imperative to define the immediate actions, and corrective and preventive actions (CAPA) in the scope of a retrospective validation. Also it is even in a team approach impossible to do all things at the same time; a risk-based prioritisation and unique predefinitions of tasks are the keys to success.

To find out what comes first and needs to be solved a root causes investigation should be done: Why was the system not validated? Before you have not answered this question a retrospective validation can not be executed successfully. Of course the retrospective validation can be executed simply following the V-model as for a prospective validation, and it can be decided to setup a parallel manual process during the validation period for verification and control, and/or the system can just be switched off until validation is finalized.

But in real terms the quality system approach must include also the corresponding remediation actions for all investigated root causes (continuous improvements).

Some examples of possible root causes are listed above, why the system was not validated:

- ➔ CSV Policy or SOPs are not in place, although defined e.g. in the Site Master File
- ➔ CSV Policy or SOPs are not followed or controlled, internal audits not executed
- ➔ CSV Policy is not known, trained, or covering relevant departments (e.g. IT)
- ➔ Lack of resources to execute the validation
- ➔ The regulatory requirement, that computerized systems should be validated, was not know – lack of awareness
- ➔ QRM process not implemented, not efficient, not trained or communicated
- ➔ Management decided to skip validation because of costs
- ➔ Procurement dept. decided to skip (external) validation because of costs
- ➔ IT systems can be purchased and released without Quality Unit approvals
- ➔ No Validation Master Plan, no inventory list in place
- ➔ Person responsible for validation left the company – no replacement assigned
- ➔ Internal IT not part of the Quality System, no procedures existing
- ➔ External IT (outsourced) does not know GXP requirements
- ➔ Lack of communication or reporting structures
- ➔ Lack of training and/or skills to validate a system
- ➔ Procedures for process validation or qualification does not include computerized systems references, or are incomplete or not sufficient
- ➔ SOP for computer system validation in place, but no forms and templates for execution available (not as living process)
- ➔ Defined SOP process for computer system validation too complex, insufficient or not appropriate

Some given root causes above may sound bizarre or not authentic. It defers to each individual's judgment as to whether these are real or fictitious examples. But looking on the different situations, conditions, aspects, impacts, and results a retrospective validation itself is not a very attractive or desired topic in general.

So based on the found root causes different actions are definitely required:

- ➔ It may start with a training of regulations on different levels
- ➔ It may required the creation of a SOP for validation
- ➔ Corrective actions: A couple of improvements might be required, e.g. for internal auditing, communication plans, notifications to Quality dept. etc.
- ➔ Corrective actions: Improvement of the Quality Risk Management process, or even a full compliance verification of all quality elements
- ➔ Corrective actions: Improvements of organisational charts, new roles and clear responsibility definitions (job roles, etc.)
- ➔ Preventive actions may also include an investigation, if the system found as not validated is the one and only or you may find more non-validated systems

The scope of a retrospective validation on the system cluster only can quickly broaden to a wider and deeper scope, corresponding to the root causes and investigation results.

Having done several retrospective validations we found out that no validation is comparable or similar to each other. So if you hear an expert saying that you should to this and that, it might be correct or not. There is no ultimate approach to retrospective validation fitting to all.

Applying the V-model for retrospective validation

You may use the V-model also for a retrospective validation as fundamental alignment. We just outline here for each step some considerations and comments:

1. Validation Plan required, system should be registered into the inventory list, referencing also to the deviation record (or change), risk assessments, investigations
2. History Report (if applicable): Historical data of the system, if possible, including an assessment and final conclusion of the system vs. process (product) stability
3. Process maps (up-to-date) and product specifications required
4. System description (if not existing, create it). It can also be included into the Validation Plan (if useful), just make sure that it can be updated also in future.
5. Some experts make it mandatory to create a User Requirement Specification. As long as you can proof that the process maps do correspond to the (already existing) system's function, we do not see the need of a URS in terms of defining requirements to an existing system. Although it is very useful to create a URS, if such process maps are not in place, because system requirements are derived from process requirements (which are hopefully known, in the ideal case also documented).
6. Gathering of technical system documentation (if existing): Try to find as much documents and records as possible. If you are lucky the engineering or IT department is following a best practice approach for requirements / technical or functional / installation / test management. Basically validation is more or less to follow best practice standards with some additional regulatory requirements and aspects. Sometimes you may find also a Software Quality Assurance Plan or similar, which is pretty much identical with a Validation Plan. If no documentation can be found the entire basic engineering or IT approach can be questioned (it seems to be a wonder that the system is running).
7. Supplier Audit (irrespective of GAMP software category): During a retrospective validation a supplier audit is not really useful, because you can not refuse the supplier anymore and your influence in terms of quality or validation is limited. Although it might be required for example to get a deeper insight into the programming standards and conventions. But pay attention that the procedures are traceable to the time period when your system was really developed. If the supplier has defined coding standard 6 weeks ago, this might not satisfy your needs at all.
8. Coding, Programming, Configuration: It is rarely possible that records of the coding or programming can be found. A retrospective validation is looking on the system more as to an as-build black-box. It should be clear that a retrospective approach compared with a prospective validation will never ever have the same quality of validation and compliance.
9. Installation Qualification (IQ): The system is already in place, but the proper installation can be verified against its current status. If there are hardware and software specification in place, this can be used as baseline for verification. Otherwise the current architecture should be documented (e.g. as part of the system description – ref. #4). In some cases, e.g. input/output lists of a PLC, you may find records, where the supplier ticked each connection during installation (no signatures existing). It should be decided on each single case, if this can be accepted or not.
10. Functional / System Testing: As for the IQ very often the problem is that verifications or testing are very often not documented (e.g. by screen-shots or similar). The requirements regarding documented evidence must be very often reduced. But it makes definitely sense to re-test all critical functions of a system, even if they are used

on a daily basis, so that it results at least to a proper and acceptable testing documentation.

11. Acceptance Test: Similar to # 10 – but in addition it should be noted, that if the system is creating e.g. a GXP record (since many years), and these records were reviewed and approved, it can be assumed that the system is accepted for its intended purpose. It depends on the results of the previous investigations if a full or partial acceptance test should be re-executed.
12. Validation Report required, including final conclusion. For a critical system (direct impact) and based on the intuition, that a retrospective approach is on a lower level of evidence as a prospective validation, it can or should be defined, that periodic evaluations/reviews of the system should be done more frequently (e.g. annual, staggered periods from 3 to 6 to 9 to 12 month and then annually)
13. Keep it validated: All procedures and processes, like defined in the "GAMP Good Practice Guides - Operation of GxP Computerized Systems" should be fully applied and implemented for a retrospective validation.

Conclusion: Retrospective Computer System Validation

The required skill levels and efforts for a retrospective validation execution are significant. It requires subject matter experts from different fields and areas. The results or actions of a retrospective validation can vary from a recall of all produced batches by the system down to just to collect some existing documentation and records. Most probably the truth lies somewhere in the middle of this both extreme positions.

Basically we started for the retrospective approach to define the ability to be proactive (change control), if the product is not on the way or at the patient, rather than reactive (deviation), if the final product has reached already the market. The proactive approach is a standard validation execution with some special considerations; also some facts may even simplify the validation and testing. The reactive approach is unfortunately in the first moment critical in all aspects and consequences. It requires a full and professional execution of a deviation process including investigations, root cause analysis, immediate actions, changes, and CAPAs.

But that is only half the truth: We can clearly recognize that if validation was not executed, and in terms of good engineering / IT practices (mapping process to system requirements to acceptance, vendor & contract mgt., operational processes, trainings etc.), such projects (systems / solutions) were initially budgeted e.g. to 1m € and ended finally at about 3m €. Also still not all process requirements are met, usability and end-user acceptance is on a low level and in addition with a very high compliance risk (business risk). It is understandable if senior management is facing all this and now additional costs for validation may come up they won't be amused at all.

At the end retrospective validation is a complex and challenging approach. Here we were just able to give an impression of it and to make no claim to be complete on this topic and different aspects. It can only produce an impulse of thought or give some ideas.

Finally it should be the ambition to avoid and eliminate any risk to patient safety and product quality. If such risks are detected, the difficulty and complexity is that this needs to be done retrospectively – and as fast as possible and with the highest level of safety, competence, transparency, and reliability.