AssemblX

A toolkit for the modular assembly of multi-gene constructs

www.assemblx.org

User manual

V1.0

08.05.2016

Index

1. The AssemblX toolkit

This toolkit allows the assembly of up to 25 transcriptional units, consisting of many more individual DNA fragments, on a single plasmid in the host *Saccharomyces cerevisiae*. The assembly strategy employs recombination-based cloning methods. These methods allow a highly flexible assembly of different DNA parts, independent of restriction sites, thereby avoiding scar sequences. Furthermore, subcloning of the multi-gene assembly product into other destination vectors for gene expression in virtually any organism is straightforward.

2. Contents

The toolkit contains a collection of Level 0, 1 and 2 vectors (**Figure 1** - **Figure 3**), which are sufficient for every assembly process using AssemblX.

2.1. Vector maps

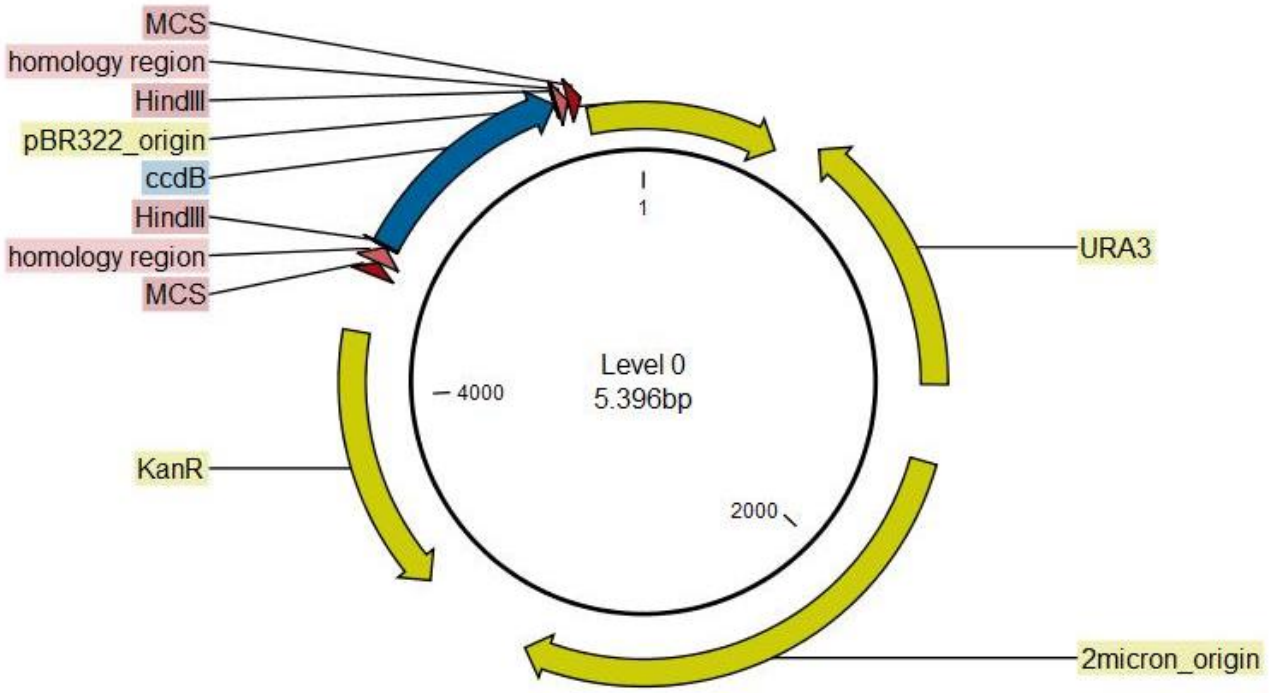
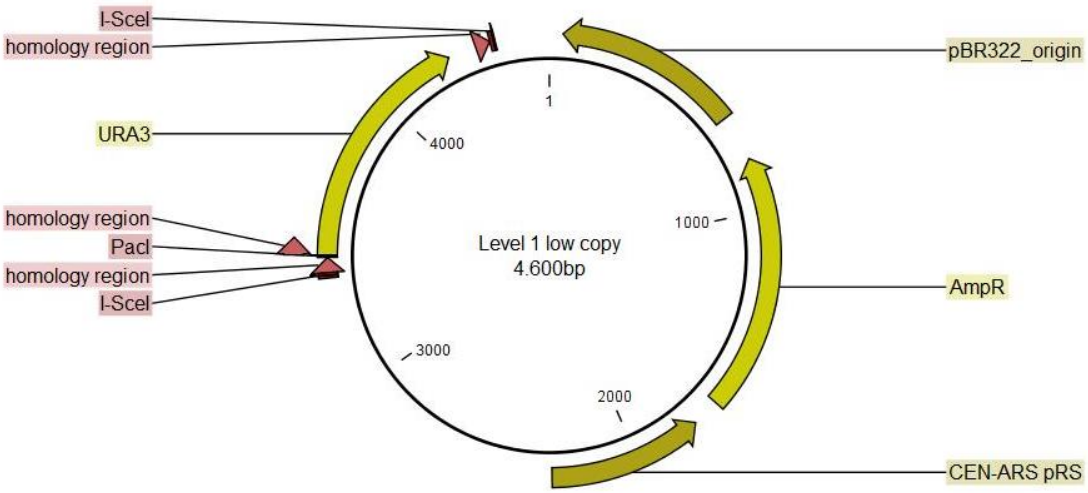


Figure 1: Level 0 vector map. High-copy version of the vector.

A



B

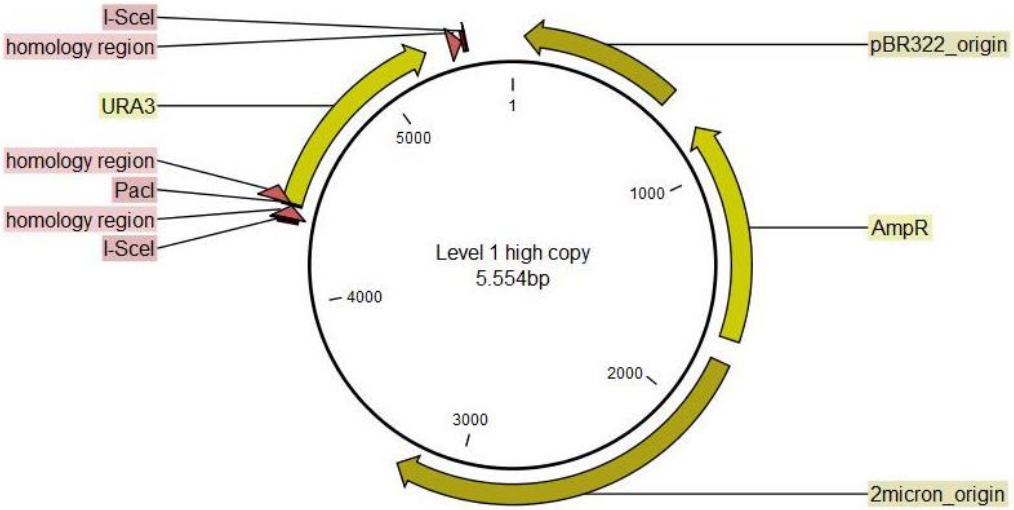
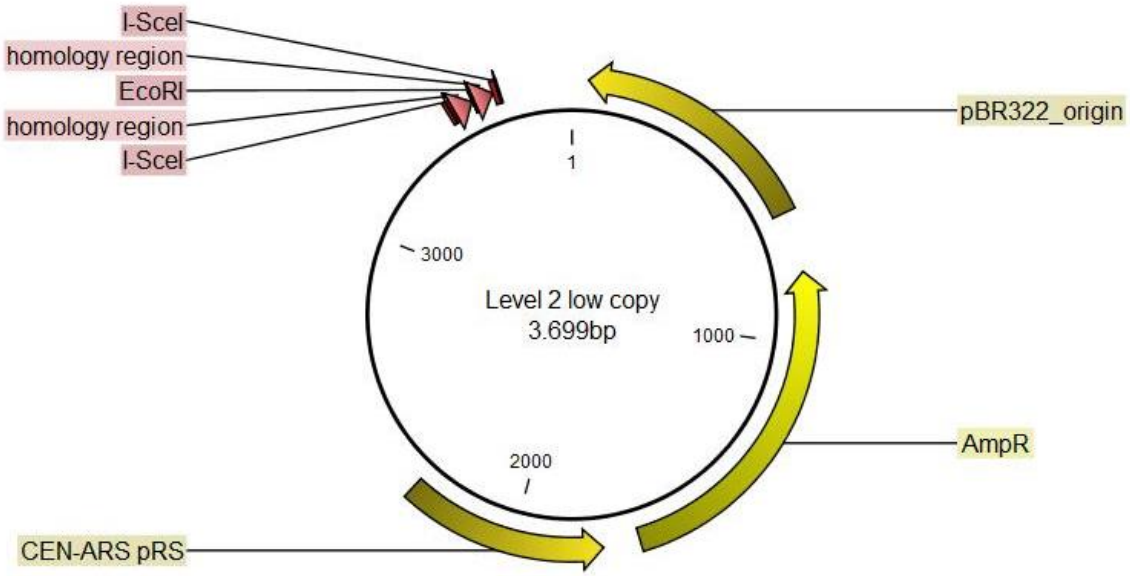


Figure 2: Level 1 vector map. A: Low-copy version of the vector. B: High-copy version of the vector.

A



B

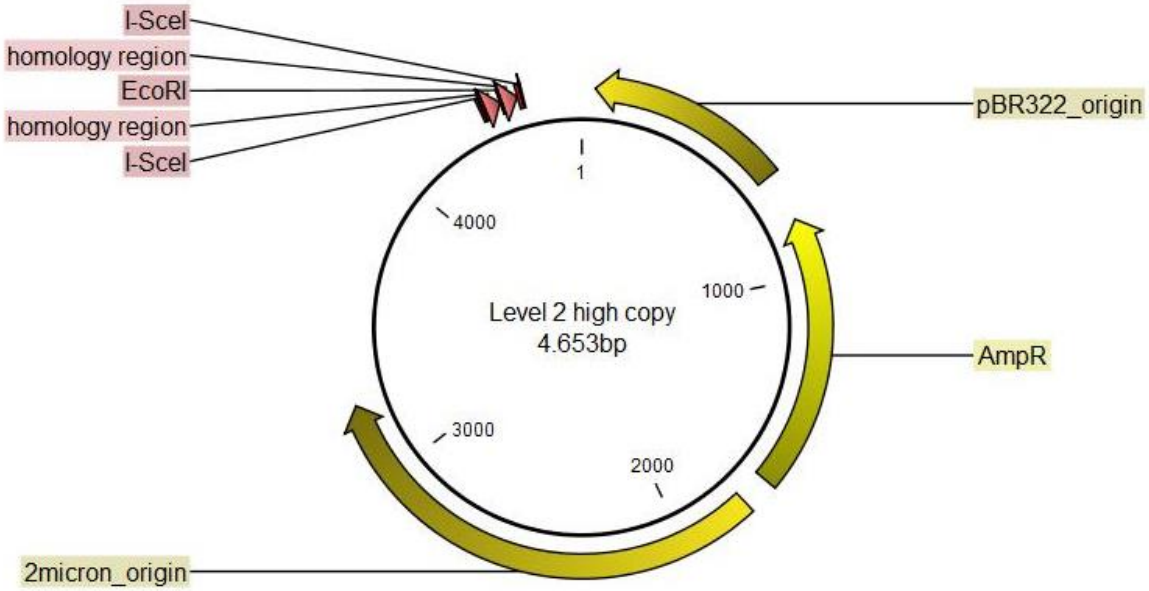


Figure 3: Level 2 vector map. A: Low-copy version of the vector. B: High-copy version of the vector.

2.2. Sequences of homology regions

The sequences of the different homology regions used for the overlap-based cloning steps, and their occurrence in the Level 0, Level 1 and Level 2 vectors are given in **Table 1 - Table 3**.

Table 1: Sequences of homology regions used to assemble transcriptional units in Level 0 vectors and to further assemble Level 0 modules into Level 1 vectors.

Name	***Sequence (5'-3')***
A0	*AAATAGGAAAGTCTGGTAACTCAGGACACTGTATCTGCTACGCTGTTTAT*
A1	*TAAAGTGAGATTCGCACAATAAACGCTGCTTCTGTT*
A2	*TAGATTCTGTGAAACTCTCAATAAAGGGACTTCGTC*
A3	*TCCTTCCCAGTGAATAGAAAAGTAGTTGCTGTTACA*
A4	*TAATAAGTTCAGTTTCGGATACCTGCCCAATAGAGT*
AR	*ATGTCGAAAGCTACATATAAGGAACGTGCTGCTACTCATCCTAGTCCTGTT*
B0	*TAAGGTTTCAGTCACACTACGGATACTTTTACAACGGGAGCAGTTATTCA*
B1	*CAATAAGTGCTATCCTGAATCTCTGGTTGTGAAACA*
B2	*AGTTGAGAATCTATTTTACTACCCGAAGACTGCGTA*
B3	*ATTACGGTCTGTGGATTTCTCACTGAACGCAATAAA*
B4	*TTGGAAGTCTACTCCCTGAACTGTGAAGAATACTAT*
BR	*ATGTCTGCCCCTAAGAAGATCGTCGTTTTGCCAGGTGACCACGTTGGTCAA*
C0	*TATTGCGTGTGATACTTCAGGAGAACCGTTATTCCGTAAAACTACAGTCA*
C1	*GGTATCGTAATCGCAACTCACTGTAAGGACTATTTA*
C2	*TAGTAAGTTATTATCGCACTCACGAGAAGCGTATTC*
C3	*ATAAACTCTGAATACTCGGGATTTCCTGGCAATAGT*
C4	*TGGTGATTCAATAAAGCGTCGTTACACAGATACTTC*
CR	*ATGACAGAGCAGAAAGCCCTAGTAAAGCGTATTACAAATGAAACCAAGAT*
D0	*CACGAGCGTTGAGGCGGATTTCAGATTATTTACAGACACTATTCTACAAT*
D1	*AAATAACTGCCACTACTGAAGATTCGGTGATTGTCT*
D2	*ACCACTGTAAAGGTATTATTCGGAAGTTCTCTACGA*
D3	*AACTGAGATTGGATAGACTACCTGTTCGTGTAATAC*
D4	*ATTTATCGGAGAAAACCTCGTCGTTGAACCAGATTT*
DR	*ATGACTAACGAAAAGGTCTGGATAGAGAAGTTGGATAATCCAACTCTTTCA*
E0	*TCTCTTACAATAAAATAGAATACTCTGCGAAGGGCGTGACACTGCTGTTT*
E1	*AGTTTATCAATCCGCTACAGGACGGTATTTCGTTAT*
E2	*TTGATACGAAGTGTGACAATCCTTTTACTGAACCGA*
E3	*TAAAAGTAAACTCTCACCCGTAGGATTGGTATCTGT*
E4	*TAGTGATAGCGTTCAATACTTCCTGGCACGATAATA*
ER	*ATGTCTGTTATTAATTTCACAGGTAGTTCTGGTCCATTGGTGAAAGTTTGC*
F0	*AAAGTATTGGTGTGAGTAATCGCTATCTCTTCTACGACTGGCTACAACAA*

Table 2: Collection of Level 0 vector sets A-E and Level 1 vectors A-E with their respective homology regions.

Level 0 Vector set	*Vector name*	*HR1*	*HR2*	*Corresponding Level 1 vectors (HR1/HR2) & marker gene*
A	pL0A_0-1	A0	A1	pL1A-hc / pL1A-lc (A0/AR) *URA3*
	pL0A_1-2	A1	A2	
	pL0A_2-3	A2	A3	
	pL0A_3-4	A3	A4	
	pL0A_0-R	A0	AR	
	pL0A_1-R	A1	AR	
	pL0A_2-R	A2	AR	
	pL0A_3-R	A3	AR	
	pL0A_4-R	A4	AR	
B	pL0B_0-1	B0	B1	pL1B-hc / pL1B-lc (B0/BR) *LEU2*
	pL0B_1-2	B1	B2	
	pL0B_2-3	B2	B3	
	pL0B_3-4	B3	B4	
	pL0B_0-R	B0	BR	
	pL0B_1-R	B1	BR	
	pL0B_2-R	B2	BR	
	pL0B_3-R	B3	BR	
	pL0B_4-R	B4	BR	
C	pL0C_0-1	C0	C1	pL1C-hc / pL1C-lc (C0/CR) *HIS3*
	pL0C_1-2	C1	C2	
	pL0C_2-3	C2	C3	
	pL0C_3-4	C3	C4	
	pL0C_0-R	C0	CR	
	pL0C_1-R	C1	CR	
	pL0C_2-R	C2	CR	
	pL0C_3-R	C3	CR	
	pL0C_4-R	C4	CR	
D	pL0D_0-1	D0	D1	pL1D-hc / pL1D-lc (D0/DR) *LYS2*
	pL0D_1-2	D1	D2	
	pL0D_2-3	D2	D3	
	pL0D_3-4	D3	D4	
	pL0D_0-R	D0	DR	
	pL0D_1-R	D1	DR	
	pL0D_2-R	D2	DR	
	pL0D_3-R	D3	DR	
	pL0D_4-R	D4	DR	
E	pL0E_0-1	E0	E1	pL1E-hc / pL1E-lc (E0/ER) *TRP1*
	pL0E_1-2	E1	E2	
	pL0E_2-3	E2	E3	
	pL0E_3-4	E3	E4	
	pL0E_0-R	E0	ER	
	pL0E_1-R	E1	ER	

	pLOE_2-R	E2	ER
	pLOE_3-R	E3	ER
	pLOE_4-R	E4	ER

Table 3: List of Level 2 vectors and their homology regions used for recombination with the Level 1 modules.

Level 1 modules used for assembly	Corresponding Level 2 vectors (HR1/HR2)	Auxotrophic markers for selection of assembled product
A,B	pL2_AB_hc/pl2_AB_lc (A0_C0)	*URA3, LEU2*
A,B,C	pL2_AC_hc/pL2_AC_lc (A0_D0)	*URA3, LEU2, HIS3*
A,B,C,D	pL2_AD_hc/pL2_AD_lc (A0_E0)	*URA3, LEU2, HIS3, LYS2*
A,B,C,D,E	pL2_AE_hc/pl2_AE_lc (A0_F0)	*URA3, LEU2, HIS3, LYS2, TRP1*

2.3. Expression data of promoter-terminator combinations

For the successful construction and expression of multi-gene pathways it is often necessary to adjust the expression level of individual genes. To provide a first starting point for researchers working with *S. cerevisiae*, we cloned a panel of constitutive yeast promoters, each in combination with the terminator region from the same gene. To easily assess the promoter strengths, we cloned promoter-yEGFP-terminator cassettes of each native promoter-terminator pair into low-copy Level 1 vector pL1-lc and transformed the constructs into *S. cerevisiae*. **Table 4** contains all primers used for amplification of promoters and terminators from yeast genomic DNA. By using flow cytometry, the relative yEGFP fluorescence for each promoter-terminator pair was determined after 16 h of batch culture. To address two widely used growth conditions, we used the carbon sources glucose or galactose for these experiments. Our results show a wide range of yEGFP fluorescence intensities mediated by the 40 different promoter-terminator pairs (**Figure 4**). For a new assembly, the selected promoter and the corresponding terminator can be amplified from the provided plasmids with the necessary overlaps for Level 0 cloning incorporated in the primers. To reduce PCR steps and primer expenses, it is also possible to amplify and clone the complete promoter-terminator cassette into a given Level 0 vector. The resulting plasmid can afterwards be linearized using *Not*I and the desired CDS can be inserted by ligation- or recombination-based cloning.

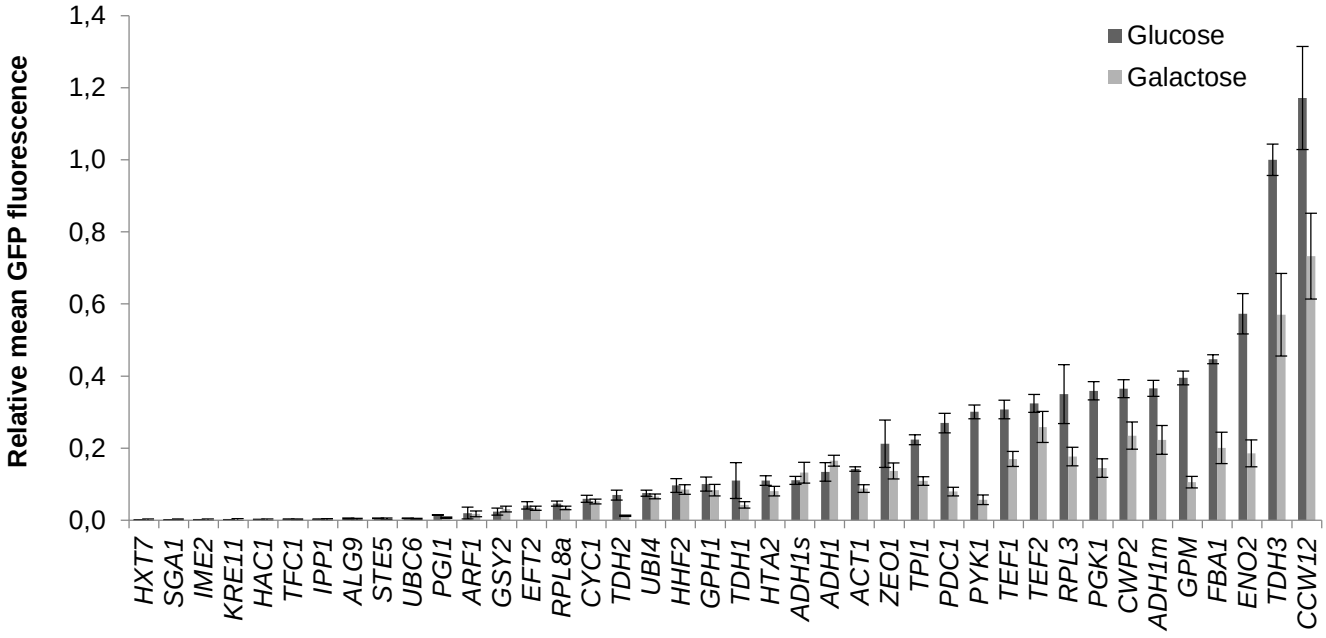


Figure 4: Comparison of 40 different constitutive promoter-terminator-pairs from *S. cerevisiae*. The mean yEGFP fluorescence per cell was measured using flow cytometry after

16 h of batch cultur in either glucose- or galactose-containing SD medium. Fluorescence values are given relative to the *TDH3* promoter-terminator pair.

Table 4: Primers used for amplification of promoters and terminators from yeast genomic DNA.

Name	***Sequence (5´-3´)***	***Target***
P614	CAACTTCTTTTCTTTTTTTTTCTTTTCTCTCTCCC	*ADH1* promoter
P615	TGAGATAGTTGATTGTATGCTTGGTATAGC	
P618	GCGAATTTCTTATGATTTATG	*ADH1* terminator
P619	GGTGTGGTCAATAAGAGC	
P620	GTAACAAAAATCACGATCTGGGTGGG	*PGI1* promoter
P621	TTTTAGGCTGGTATCTTGATTCTAAATCG	
P624	ACAAATCGCTCTTAAATATATACCTAAAGAAC	*PGI1* terminator
P625	GTAGTTTAGTGTTTTTCTTCCAGTGCG	
P626	ATTGGTTTTTCCAGTGAATGATTATTTGTCGT	*TDH2* promoter
P627	TTTGTTTTGTTTGTTTGTGTGATGAATTTAATTTGA	
P630	ATTTAACTCCTTAAGTTACTTTAATGATTTAG	*TDH2* terminator
P631	GGCGAAAAGCCAATTAGTGTG	
P632	ACTTCTCGTAGGAACAATTTCGGG	*HXT7* promoter
P633	CATTTTTTGATTAAAATTAAAAAAACTTTTTGTTT	
P636	TTTGCGAACACTTTTATTAATTCATGATCACGC	*HXT7* terminator
P637	AATAACTGACTCATTAGACACTTTTTGAAGCGGG	
P638	ACGCACAGATATTATAACATCT	*PGK1* promoter
P639	TGTTTTATATTTGTTGTAAAAAGTAGA	
P642	ATTGAATTGAATTGAAATCGATAGATCA	*PGK1* terminator
P643	AGGCATTAAAAGAGGAGCG	
P644	GGGGCCGTATACTTACATATAGTAGATG	*TEF2* promoter
P645	GTTTAGTTAATTATAGTTCGTTGACCGTATATTC	
P648	GAGTAATAATTATTGCTTCCATATAATAT	*TEF2* terminator
P649	GGGGTAGCGACGGATTAAT	
P650	AATGCTAGTATTTTGGAGATTAATCTCAGTACA	*PYK1* promoter
P651	TGTGATGATGTTTTATTTGTTTTGATTGGTGTCT	
P654	AAAAAGAATCATGATTGAATGAAGATATT	*PYK1* terminator
P655	GCATTTATGTACCCATGTATAACCTTCC	
P656	GTGTCGACGCTGCGGGTATAG	*ENO2* promoter
P657	TATTATTGTATGTTATAGTATTAGTTGCTTGGTG	
P660	AGTGCTTTTAACTAAGAATTATTAGTCTTTTCTGC	*ENO2* terminator
P661	ATCTCCATCTCCCATATGCATATCACTGTGG	
P662	CATGCGACTGGGTGAGCATATGTTCCG	*PDC1* promoter
P663	TTTGATTGATTTGACTGTGTTATTTTGCGTGAGG	
P666	GCGATTTAATCTCTAATTATTAGTTAAAGT	*PDC1* terminator
P667	GGCAGTTTTGAATTGAGTAACCA	
P668	ATAACAATACTGACAGTACTAAATAATTGCC	*FBA1* promoter
P669	TTTGAATATGTATTACTTGGTTATGGTTATATATGA	
P672	GTTAATTCAAATTAATTG	*FBA1* terminator
P673	GCTATCAAAAACGATAGATC	
P674	AGTTTATCATTATCAATACTGCCATTTCAAAG	*TDH3*

Name	*Sequence (5´-3´)*	*Target*
P675	TTTGTTTGTTTATGTGTGTTTATTCGAAAC	promoter
P678	GTGAATTTACTTTAAATCTTGCA	*TDH3* terminator
P679	GGAATCTGTGTATATTACTGCATCT	
P680	TAGTCGTGCAATGTATGACTTTAAGATTTG	*GPM1* promoter
P681	TATTGTAATATGTGTGTTTGTTTGGATTATTAAG	
P684	GTCTGAAGAATGAATGATTTGATGATTTCTTTTTCC	*GPM1* terminator
P685	TATTCGAACTGCCCATTCAGC	
P686	TATATCTAGGAACCCATCAGG	*TPI1* promoter
P687	TTTTAGTTTATGTATGTGTTTTTTGTAG	
P690	GATTAATATAATTATATAAAAATATTATCTTCTTT	*TPI1* terminator
P691	TATATAACAGTTGAAATTTGGATAAGAACATC	
P692	ATAGCTTCAAAATGTTTCTACTCCTTTTT	*TEF1* promoter
P693	TTTGTAATTAAAACTTAGATTAGATTGCTATGCT	
P696	GGAGATTGATAAGACTTTTCTAGTTGC	*TEF1* terminator
P697	AGATAGCGCCGATCAAAGTATTTG	
P698	ACAAGCGCGCCTCTACCTTG	*ACT1* promoter
P699	TGTTAATTCAGTAAATTTTCGATCTTGGG	
P702	TCTCTGCTTTTGTGCGCGT	*ACT1* terminator
P703	TACACGGTCCAATGGATAAACATT	
P704	AGCGTTGGTTGGTGGATCAAGCC	*CYC1* promoter
P705	TATTAATTTAGTGTGTGTATTTGTGTTTGTGTGTC	
P708	ACAGGCCCCTTTTCCTTTGTCGA	*CYC1* terminator
P709	AGCTTGCAAATTAAAGCCTTCGAGCGT	
P710	GATACCTATTCTCTCAGTAGCTGTTTCACC	*IME2* promoter
P711	AAATGACCTATTAAGTTAAGCTTAGTACTCTTC	
P714	TAATTTGTCATTTGACTGACGAATCGT	*IME2* terminator
P715	TCGTATGCAGTAATAGGGTTTTTTGAGAATATACCG	
P716	TTAGTAAAAATGTGCGCACCAC	*TDH1* promoter
P717	TTTGTTTTGTGTGTAAATTTAGTGAAGTAC	
P719	ATAAAGCAATCTTGATGAGGATAATGATTT	*TDH1* terminator
P720	TATTGATCCGTACTAATATTGTTAGATGAGCTGC	
P721	GAAGTTTCACTACCCTTTTTCCATTTGCC	*ADH1* promoter (s)
P615	TGAGATAGTTGATTGTATGCTTGGTATAGC	
P722	ACAATATGGAAGGGAACTTTACACT	*ADH1* promoter (m)
P615	TGAGATAGTTGATTGTATGCTTGGTATAGC	
P723	TAGAAGGCGTATTGCTCAATAGTAGAAAGCAGGC	*STE5* promoter
P724	TTAAAAGTTGTTTCCGCTGTATCCTGTATCTTCC	
P727	AGTATACACTAAATTTTATGCAATAATAAAAAGA	*STE5* terminator
P728	AATAATATTGACCCACCTTCTGCTCCGG	
P729	GTTGGAATGAAATTTTAATATCATCTATTTCGCAGC	*IPP1* promoter
P730	TAGTAAATTGCGCGGCGGG	
P733	AATATTTTGAATTGAAAATGAGACACCTACCACG	*IPP1* terminator
P734	TTTGTTGATTGTCATCAGTTTTAGTAAAAAA	
P735	ACGCTCAACGGAAGAGGAAG	*GPH1* promoter
P736	TGTTCAAAATTAAATTAAGTTGAAAGCTGCT	
P739	CTGGGAGTTTTTTTTATCTCTTTGGCACCC	*GPH1*

Name	*Sequence (5´-3´)*	*Target*
P740	GTTATACTATTCTCTCTTCCTTTCTGGCCTCG	terminator
P741	TAGAGTGACCGATGAGTTGCATG	*GSY2* promoter
P742	TCAAAATTTTTCTCTGAGGTAGTCAAAAC	
P745	ATCCTATGAGGATATAAACAGTATTAAAAAAATCT	*GSY2* terminator
P746	CGACCGCCTTAATTCAAAATTTATCTTTCGT	
P747	TATACAGCTTGTGATTTCGCGCGCC	*SGA1* promoter
P748	TTTGCTGCTGTTACTTATTTGAAATCTTGC	
P751	ACAAAAAAAAAAATAAAAGGAAAGCGAGAAG	*SGA1* terminator
P752	TATCTAATCAATTTATAATATCAGAGTTTTGT	
P753	GACCTACCTTAAAACACGTTTCGTTTAAG	*ALG9* promoter
P754	CGTTCCCTTATAATTGTGCTTCAGGG	
P757	TATACAGGTTCTCTAAAACAAGCAGC	*ALG9* terminator
P758	GGCGTTGGACATCCATGAC	
P759	ATTCAACGAAACTTACGACAAGATAGTCTTTGG	*UBC6* promoter
P760	TACTATTGTACGTACTTTGTTTGCAATTTGC	
P763	AGCCATTAAATGAACTAAACTTTAGATAAATTTTG	*UBC6* terminator
P764	AATGGGAGTTTCTGTCTCCCTAGGT	
P765	CCACTTTTTGTCTTATGAGTGCTTTCTTGTATATGC	*KRE11* promoter
P766	GGTCAATGAGTTGTGATTTATCAATTGCGC	
P769	GCGGAACAAGTATACATTTTATTAACATAGAC	*KRE11* terminator
P770	CGCAACGCCTAAGTAAAGCTG	
P771	AAATTATTTTTCTTCTTCCTTTTTCTCATACATATC	*TFC1* promoter
P772	TTTCACGATTTGGAAGATATTTTCGC	
P775	TAAGTAGGTTTTCTTTTATTTCAATATCAAAA	*TFC1* terminator
P776	CCGTTCTTTCTGGAGGCATTCT	
P808	CTAATAGACAAGGTGCTATGAGTGAA	*CWP2* promoter
P809	TTTTTTTCTTGTTAGTGTGTAGCGAA	
P810	GAAATCTCTGATTTTTTATAATATCTATATGGCT	*CWP2* terminator
P811	AAAAAGATCCCACCGTAGCACACC	
P812	CCTTTCCTGCGGGATAGCTTTTGC	*RPL3* promoter
P813	GATTGATTGTTGTAGTAACTGTGTTGTTC	
P814	GAAGTTTTGTTAGAAAATAAATCA	*RPL3* terminator
P815	GCAATATAATGGACGGGT	
P816	CAACATAAATAATTTCTATTAACAATGTAATTTCCA	*RPL8a* promoter
P817	TTCGAATTAGTTGTTTTGATGTGATAATAGAGGG	
P818	ATTGAAAATGAGAAATTTTGCATAAAAAA	*RPL8a* terminator
P819	AATGATCGTGGAGTTTCAAACATCG	
P820	AACCAGGGCAAAGCAAAATAAAAG	*CCW12* promoter
P821	TATTGATATAGTGTTTAAGCGAATGACAGAAG	
P822	ACTTAGTTTATTATTATTTATACATTCTAAATTT	*CCW12* terminator
P823	ATGCTATGTAATAGACAATAAAACCATGT	
P824	TGTGGAGTGTTTGCTTGGATCC	*HHF2* promoter
P825	TATTTTATTGTATTGATTGTTGTTTTTGCTACTC	
P826	ACAATCGGTGGTTAAACAATCG	*HHF2* terminator
P827	TTTCTTTTATTGAGACTTATTTTTAATATATATAG	
P828	TCTCAAATCCATTGGTGTTTCCATTAGGCC	*EFT2*

Name	*Sequence (5´-3´)*	*Target*
P829	CTTGCAATTATGTTATAGTTTGTGTTTGTGG	promoter
P830	GAATGGTTAAACAATTTTTAATTATT	*EFT2* terminator
P831	GCTTTCATGCTAGGAATGTG	
P832	TAGTTGTAGAGTAAGTTGTTGTTGGTTTG	*HTA2* promoter
P833	TATATATTAAATTTGCTCTTGTTCTGTACTTTCC	
P834	GAACTGAGTTGAAAAGAAACAAAGCA	*HAT2* terminator
P835	CTGTAATATATCTTTATAACATGTATACTAATGA	
P836	TTTTAGAGTCTTCCTTGACCAATCATCTTATTCG	*UBI4* promoter
P837	AATCTATTAGTTAAAGTAAAGTGGGAGGG	
P838	TCAGTCCTCGCAATATTTTCATTATGTC	*UBI4* terminator
P839	TTAACTGTTCTTAATTTTCCTTTTTTCTGCC	
P840	CTGTCTTCCTCTACTGGGCTTATC	*HAC1* promoter
P841	AGTGGCGGTTGTTGTCGTAGG	
P842	AGACAATCGCAAGAGGGTATAATTTTTTTTTTCTCG	*HAC1* terminator
P843	CTCATTGGCCCGGAGTTAGGGGG	
P844	TACTTAAGAGCTAAACGAAAAGATAAGTATCTCGC	*ARF1* promoter
P845	TTTTAATTCTATTAAACCTGTTTGAGTTCTTTC	
P846	AAATTCTAGAATATGGATCAAATACGCTTG	*ARF1* terminator
P847	ATTAATACAGTCCCCAGATTGAGAAAG	
P849	TTAAACGTGTGGTTTATGGGTGCACCAGGG	*ZEO1* promoter
P850	TATTAATTGATATAAACG	
P851	TCACACTTGAATGTGCTGC	*ZEO1* terminator
P852	TATTTTTACTTTCTTATCAAACTTTTTTTCCTCCGG	

3. Planning the assembly

The toolkit enables the user to assemble up to 25 transcriptional units, consisting of many more DNA fragments in total, in a single vector for expression in the host *S. cerevisiae*. Furthermore, the subcloning of multi-gene assemblies into destination vectors for the transformation of virtually any organism is possible and straightforward.

The destination vector used for the integration of the fully assembled gene array is called xYAC. The vector allows the expression of multiple genes in *S. cerevisiae*; it is a circular vector with either a high-copy or a low-copy replication origin. The generation of destination vectors for other target organisms is described in Section 4.3.4.

The xYAC assembly relies on recombination-based cloning methods. These are SLiCE (Zhang et al., 2012), NEBuilder® HiFi DNA assembly (a variant of Gibson assembly) (Gibson, 2011) and TAR (Larionov et al., 1996) (see Section 5). These methods allow the highly flexible assembly of DNA fragments from diverse sources (no sites for restriction enzymes are required) without the introduction of scar sequences and without the need for laborious optimization of DNA sequences such as e.g. the deletion of restriction sites in a given construct.

The assembly of up to 25 transcriptional units into a single destination vector is organized in three cloning hierarchies, for which we developed level-specific assembly plasmids: Entry Level 0, and Assembly Levels 1 and 2 (**Figure 5**).

To start the assembly, each single transcriptional unit is assembled in one of several Level 0 vectors. The assemblies are performed in a highly flexible manner by recombination-based cloning (**Figure 5**). Transcriptional units, generally composed of a promoter, a coding sequence (CDS) and a terminator, are either cloned from already existing expression cassettes or are assembled from individual parts, derived from different sources. The term ´transcriptional unit´ is used throughout the manual, although it is equally possible to assemble DNA fragments different from those mentioned above. The parts cloned into a transcriptional unit do not have to fulfill any strict requirement; a detailed protocol for cloning individual DNA parts into a transcriptional unit is given in Section 4.3.1.

Each individual Level 0 vector provides a combination of two ´homology regions´, which flank the assembled transcriptional unit. These homology regions will later direct the assembly of multiple transcriptional units in a Level 1 vector (**Figure 5**).

To this end the Level 0 vectors are divided into five sets called A, B, C, D, and E. After assembly, each set of Level 0 vectors carries up to five transcriptional units, which are then assembled into one of various Level 1 vectors. The first and last vector in each set (e.g. pL0A_0-1 and pL0A_4-R in the Level 0 A set; **Figure 6**) provide a region compatible to the Level 1 vector (homology regions A0 and AR show homology to Level 1 vectors pL1A-hc or pL1A-lc, in this example) and a region compatible to the neighboring Level 0 vector from the same set (pL0A_1-2 and pL0A_3-4, in this example). The other vectors have homology regions compatible to each other (e.g., pL0A_1-2 to pL0A_2-3). The compatibility of individual Level 0 vectors can be deduced from **Table 2**. For example, vector pL0A_1-2 provides homology regions A1 and A2, which make pL0A_1-2 compatible with e.g. pL0A_0-1, providing homology regions A0 and A1 (where A1 from pL0A_0-1 is used for recombination with A1 from pL0A_1-2) and pL0A_2-3, providing A2 and A3 (where A3 from pL0A_2-3 is used for recombination with A3 from pL0A_2-3). Hereby, the Level 0 vectors define the order of the transcriptional units in the Level 1 assembly. The homology regions are flanked by a multi-cloning site (MCS) harboring sites for five different 8-mer cutter restriction enzymes (i.e., *Swa*I, *Asc*I, *Pme*I, *Sbf*I, *Fse*I), thereby allowing the release of the transcriptional units and their further cloning into a Level 1 vector via recombination-based cloning (**Figure 6**). Details on this procedure are given in the protocol Section 4.3.1.
We generated ten different Level 1 vectors: pL1A, pL1B, pL1C, pL1D and pL1E - each corresponding to one of the five Level 0 sets. Five vectors from the Level 0 A set can be combined in Level 1 vector pL1A. Five vectors from the Level 0 B set can be combined in Level 1 vector pL1B, and so on. The assembled parts are called Level 1 modules A, B, C, D or E. Each Level 1 vector is available in a low-copy and a high-copy version.

The different Level 1 modules can be further combined into one Level 2 vector. Similar to the Level 0 vectors, the Level 1 vectors provide homology regions flanking the assembled module. These regions allow recombination-based assembly between different Level 1 vectors. Level 1 vector pL1A possesses a homology region compatible to the Level 2 vectors and to Level 1 vector pL1B. Vector pL1B shows homology to pL1A and pL1C, and so on. The last Level 1 vector has homology to Level 2. The homology regions of Level 1 modules are

flanked by homing endonuclease sites (I-*Sce*I) which allow the release of the modules and further cloning into Level 2 by TAR *in vivo* assembly in *S. cerevisiae* or another homology based cloning method (**Figure 7**; for a detailed description, see Sections 4.3.2/4.3.3).

Here again, different Level 2 vectors with different homology regions are provided to allow the combination of two, three, four or even five Level 1 modules. Therefore, eight different Level 2 vectors are provided. The Level 2 vector pL2_AB allows assembly of modules from Level 1 A and B. Level 2 vector pL2_AC is compatible with the assembly of the modules from Level 1 A, B, and C, and so on.

Subcloning of the assembled multi-gene fragment into a destination vector for the transformation of any organism is possible by insertion of an I-*Sce*I-*ccdB*-I-S*ce*I cassette. For more details, see Section 4.3.4.

Before beginning an individual assembly, it is important to plan and optimize the final construct. There are a few general points to consider, as outlined in the following chapters.

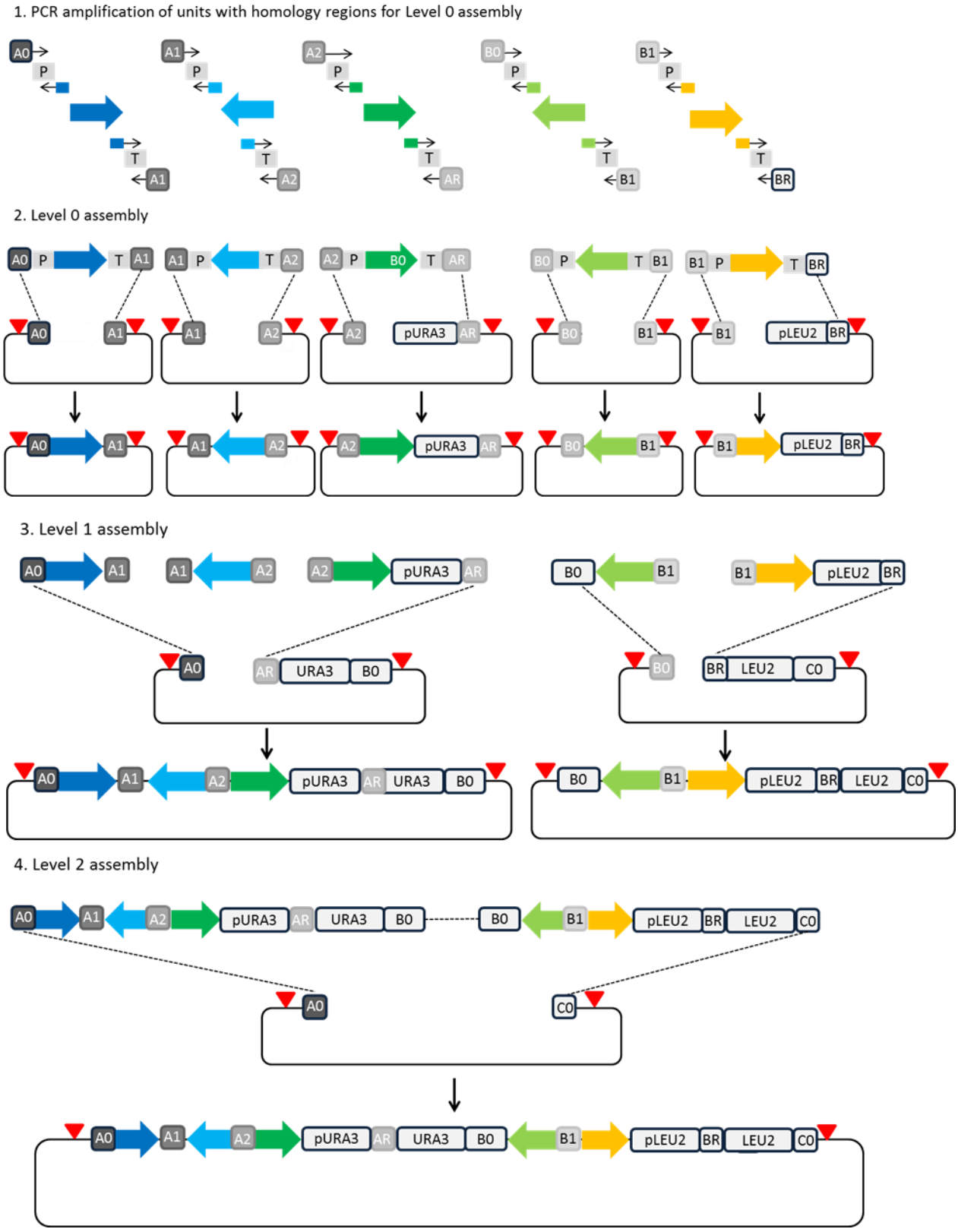
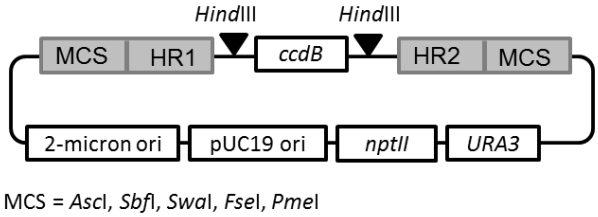


Figure 5: Overview of the assembly strategy. (1) and (2) Transcriptional units are assembled in individual Level 0 vectors. (3) Level 0 units are assembled in Level 1 vectors. Different Level 1 vectors are compatible with Level 0 vectors from different sets. (4) Assembled Level 1 modules from different Level 1 vectors are combined in a single Level 2 vector. Promoters for marker genes are designated pURA3, or pLEU2, while the promoterless CDS are named URA3 and LEU2.

3.1. Choose of the right combination of Level 0, Level 1 and Level 2 vectors

The kit includes different sets of entry vectors (Level 0); each cloning procedure starts with these Level 0 vectors. We generated five sets of Level 0 entry vectors, called set A, B, C, D, and E. Each set includes nine vectors, e.g. vectors A0-A1, A1-A2 etc. of set A, and vectors B0-B1, B1-B2 etc. of set B, and so on. The vectors differ in their homology regions (called A0, A1, … B0, B1 etc.) used for the assembly into the vector of the next level (**Figure 6**). The nine vectors of every set have compatible homology regions to each other and to one of the Level 1 vectors. Each Level 0 vector is designed to hold one (transcriptional) unit (usually composed of promoter + CDS + terminator) of the final assembly product and transcriptional units cloned into one Level 0 vector set will later be assembled into one Level 1 module.



Homology regions of:

Level 0 set A		Level 0 set B		Level 0 set C		Level 0 set D		Level 0 set E	
A0-A1	A0-AR	B0-B1	B0-BR	C0-C1	C0-CR	D0-D1	D0-DR	E0-E1	E0-ER
A1-A2	A1-AR	B1-B2	B1-BR	C1-C2	C1-CR	D1-D2	D1-DR	E1-E2	E1-ER
A2-A3	A2-AR	B2-B3	B2-BR	C2-C3	C2-CR	D2-D3	D2-DR	E2-E3	E2-ER
A3-A4	A3-AR	B3-B4	B3-BR	C3-C4	C3-CR	D3-D4	D3-DR	E3-E4	E3-ER
A4-AR		B4-BR		C4-CR		D4-DR		E4-ER	

Figure 6: General design of Level 0 vectors. A Level 0 vector with a *Hind*III-flanked *ccdB* cassette between the two homology regions and the multiple cloning sites (MCS) is shown. The different combinations of homology regions in each of the five different Level 0 sets A, B, C, D, and E are shown. The grey boxes represent the homology regions of the nine vectors of every Level 0 set. The combinations of homology regions indicated on the left side of each box are needed to clone five transcriptional units into Level 1. The combinations indicated on the right side of each box are needed for cloning of two (e.g., A1-AR), three (e.g., A2-AR) or four (e.g. A3-AR) transcriptional units into Level 1.

Up to five Level 0 modules can be assembled into one Level 1 vector. Up to five Level 1 modules can then be assembled into a Level 2 vector (**Figure 7** shows an example for six transcriptional units), which then contains up to 25 transcriptional units. The Level 2 vector can serve as the destination vector. Alternatively, the whole assembly cassette can be transferred from the Level 2 vector into a destination vector of choice using the homing endonuclease sites for I-*Sce*I provided in the vector backbone (**Figure 8** and Section 4.3.4).

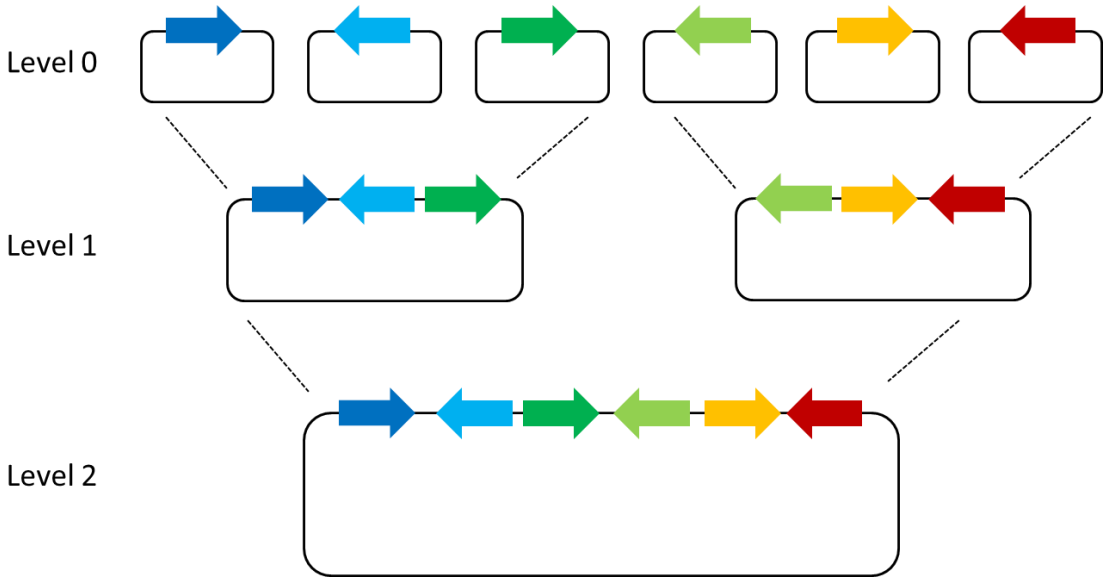


Figure 7: Level 2 assembly starting from six Level 0 vectors. Assembly of six Level 0 entry vectors from two sets into two Level 1 vectors. Further assembly of two Level 1 vectors, each containing three Level 0 units, into a Level 2 vector. Orientation of transcriptional units is suggested to be head-to-head and tail-to-tail.

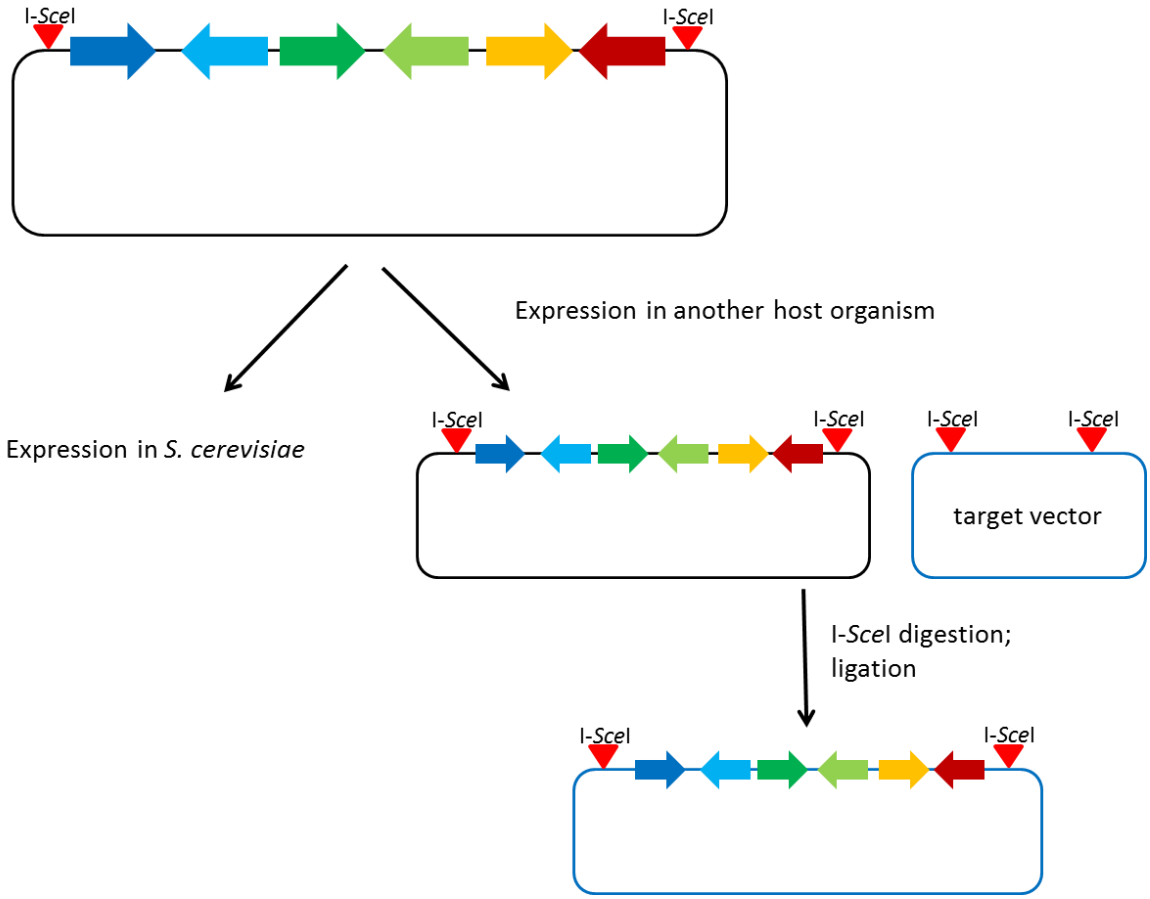


Figure 8: Subcloning of Level 1 or 2 constructs into any target vector. The fragment assembled in Level 1 or 2 vectors can be released by I-*Sce*I-digestion and ligated into another target vector (containing I-*Sce*I sites) for any host organism.

Cloning of up to five transcriptional units

For cloning of up to five transcriptional units into one xYAC, each unit has to be cloned into a Level 0 vector. Therefore, the five Level 0 vectors from set A (with homology regions compatible to Level 1 A) can be used to assemble the units (**Figure 9**). After successful assembly, the transcriptional units in the different Level 0 vectors are excised with one of the 8-mer cutters cleaving within the MCS and cloned into a *Pac*I-digested Level 1 vector via recombination-based cloning. For cloning of less than five genes it is important that the Level 0 vector holding the last unit of the assembly provides a homology region compatible to a Level 1 vector (e.g., for an assembly of three Level 0 vectors in Set A, choose pL0A_0-1 to hold the first unit, pL0A_1-2 for the second unit, and pL0A_2-R for the last unit; **Figure 10**).

For up to five transcriptional units, the Level 1 vector can serve as the final destination vector because it can be maintained and selected for in *S. cerevisiae*. Level 1 and Level 2 backbones are similar and differ only in their homology regions. Hence, in such a case subcloning into a Level 2 vector is not needed. If the final construct shall be expressed in a host organism different from yeast, the fully assembled cassette can be cloned to a compatible vector via I-*Sce*I homing endonuclease sites (Section 4.3.4).

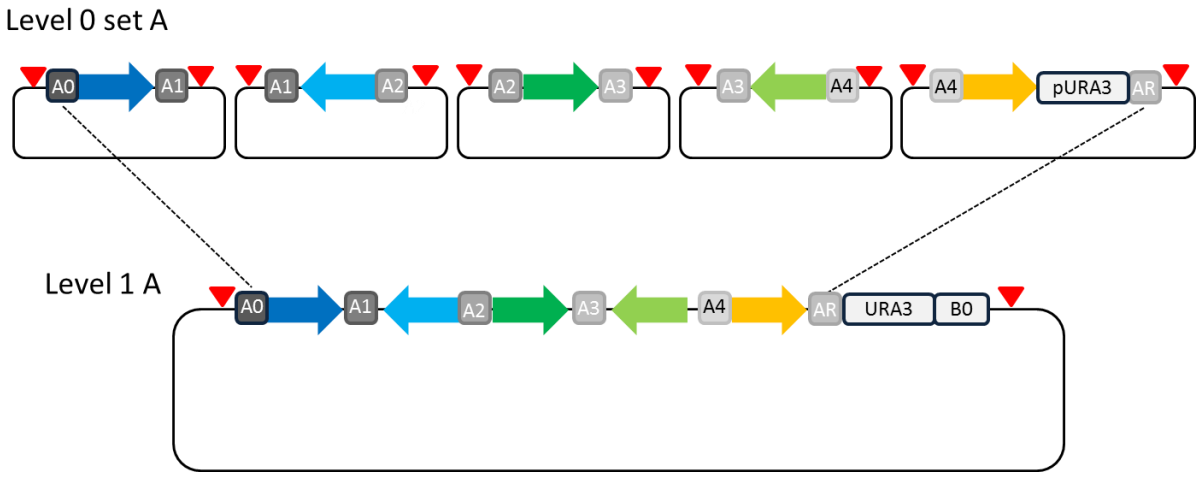


Figure 9: Cloning of five Level 0 units into a Level 1 A vector. After releasing the transcriptional units from Level 0 vectors, the fragments are assembled into a Level 1 vector via flanking homology regions, using recombination-based cloning methods. The first and last vector of each Level 0 set (e.g., vectors A0-A1 and A4-AR) possess homology to the appropriate Level 1 vector (e.g., vector A0-AR). Homology regions of vectors in between determine the order of the Level 0 vectors.

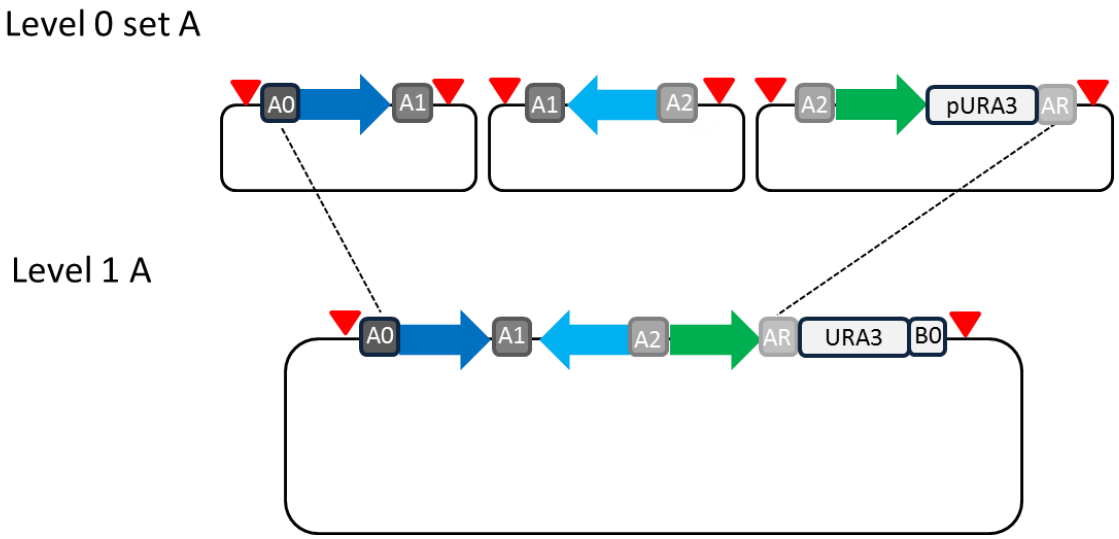


Figure 10: Cloning of three Level 0 units into a Level 1 A vector. After releasing the transcriptional units from Level 0 vectors, the fragments are assembled into a Level 1 vector via flanking homology regions, using recombination-based cloning methods. The last Level 0 vector needs to have a homology region fitting to Level 1 A to allow for the correct assembly of the Level 1 module.

Cloning of more than five genes

To assemble more than five genes, the different sets of Level 0 vectors are used. For example, to clone ten transcriptional units, the vector sets Level 0 A, and B are employed. Five units are cloned in Level 0 A, and five in Level 0 B. Afterwards, the Level 0 A set can be assembled into a Level 1 A vector, and the Level 0 B set into a Level 1 B vector, to give Level 1 A and B modules, which will be combined into one Level 2 vector. The last Level 1 vector used determines the Level 2 vector required for the assembly. If the last vector is Level 1 B, a Level 2 B vector must be used to assemble the Level 1 modules A and B.

Two options for cloning a number of units that cannot be divided by five:

1. For the example of cloning 13 transcriptional units two consecutive sets of Level 0 vectors (Level 0 A and 0 B) are filled up with five units each and the remaining three units are cloned into vectors of the next Level 0 set (Level 0 C in this example). The Level 0 vector holding the last unit of each set has to carry a homology region compatible to the appropriate Level 1 vector. For the Level 0 Set C these would be vectors providing the CR homology region. For example:

   Assembly with 13 genes:

   - Five genes -> Level 0 A
   - Five genes -> Level 0 B
   - Three genes -> Level 0 C (C0-C1; C1-C2; C2-CR)

2. Distribute the units over any number of Level 0 sets. Each set may contain between one and five genes. Be aware, that only consecutive vector sets can later be

assembled in Level 2. This means, that an assembly of a Level 1 A module together with a Level 1 C module is not possible. The modules have to follow the alphabetic order of vector sets A, B, C, D, and E, always starting with A. See Table 2 for more information on the homology regions provided in Level 1 vectors.

Assembly with 12 genes:

- Three genes in Level 0 A (A0-A1, A1-A2, A2-AR)
- Five genes in Level 0 B (B0-B1, B1-B2, B2-B3, B3-B4, B4-BR)
- Four genes in Level 0 C (C0-C1, C1-C2, C2-CR)

Notes:

- *Consider the orientation of each transcriptional unit within the final assembly product. In the final assembly a high number of genes/gene regulatory elements may be placed in close proximity to each other. To minimize the probability of undesired effects, carefully plan the order and relative orientation of each transcriptional unit. We generally prefer a tail-to-tail orientation of neighboring genes. In some cases, it might be advantageous to incorporate insulator elements between individual transcriptional units.*
- *In the last step of the assembly, up to five Level 1 modules are recombined to create a Level 2 vector. In particular cases, e.g. for pathway engineering, it might be necessary to exchange individual transcriptional units of the construct at a later time point. For such cases we recommend to combine the transcriptional units (Level 0) in question into one Level 1 module, and all other units in one or more different Level 1 vectors. This will reduce the number of assembly steps needed to create the Level 2 vectors carrying the various gene assemblies: only Level 0 vectors belonging to one Level 1 module need to be modified and reassembled into the Level 1 vector.*

3.2. Choosing promoters for expression in *S. cerevisiae*

The toolkit provides a selection of native combinations of constitutive promoters and terminators for gene expression in *S. cerevisiae* (**Figure 4**). Production of the reporter gene yEGFP under different conditions was determined for each promoter/terminator pair and can be used as a measure for promoter strength. Using a certain promoter-terminator-pair for the assembly, parts can be amplified directly from the provided plasmids, using the primer design guidelines described in Section 4.

To use promoters different from those provided with this toolkit, follow the same guidelines and amplify the parts from any suitable source DNA, or produce the needed DNA fragment by gene synthesis.

3.3. Choosing the destination backbone

The assembled multi-gene constructs are suitable for transformation into different host organisms. If the host organism is *S. cerevisiae,* gene expression is possible right after the assembly in a high- or low-copy version of a Level 1 or 2 vector. If the host organism is different from *S. cerevisiae* follow one of the following two strategies:

Modify a standard expression vector to make it compatible with the assembly strategy by

- inserting two I-*Sce*I sites into a site suitable for restriction enzyme-based cloning (for details see Section 4.3.4).
- inserting two homology regions into a site suitable for overlap-based cloning (for details see Section 4.3.4).

4. Protocol section

4.1. General advice on the preparation of parts, compatible with recombination-based cloning

The AssemblX multi-gene assembly strategy relies on homology regions for recombination-based cloning via SLiCE, NEBuilder HiFi DNA assembly, or TAR. For AssemblX cloning, homology regions are added to inserts of interest to make them compatible with the upstream and downstream parts of the planned assembly. This is usually done by PCR.

PCR amplification of inserts with compatible homology regions:

- To add homology regions to your insert of choice, design primers containing 15-36 bases at their 5´-ends homologous to the sequences of the preceding/following parts in your assembly, depending on the overlap-based cloning method.
- The 3´-end of each primer (15-24 bp in length) is specific to your fragment of choice (**Figure 11**). Accordingly, the final length of primers is 30 to 60 bases.
- The automated AssemblX webtool, which is based on the design software j5 (Hillson et al., 2011) helps to design primers for transcriptional unit assembly in Level 0 vectors.
- Primers can be ordered as standard desalted primers, it is usually not necessary to use HPLC-purified primers.
- PCR products, visible as a single specific band after agarose gel electrophoresis, can be used in assembly reactions without further purification, although it is sometimes advantageous to column-purify the products.

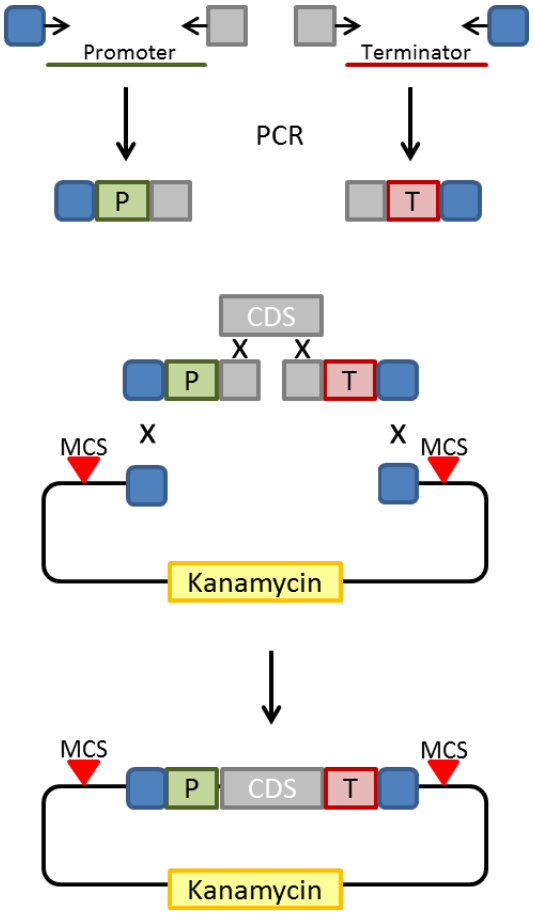


Figure 11: Assembly of a transcriptional unit in a Level 0 vector. PCR amplification of promoter and terminator with primers introducing the homology regions to Level 0 or the CDS, and subsequent cloning of three fragments in a Level 0 vector by overlap-based cloning. Note, that no extra homology regions are added to the CDS.

Use of stitching or clipping oligonucleotides for assembly reactions:

An alternative method for adding homology regions to the assembly parts uses stitching oligonucleotides. To stitch two fragments **not** sharing end homology, order 72 bases long sense and antisense oligonucleotides, containing 36 bases of each of the two fragments to be joined (**Figure 12**).

- For better results, we recommend to anneal the single-stranded oligonucleotides before use, although this is not strictly required.

Note: In our hands, stitching is not as efficient as assembly of fragments sharing end homology.

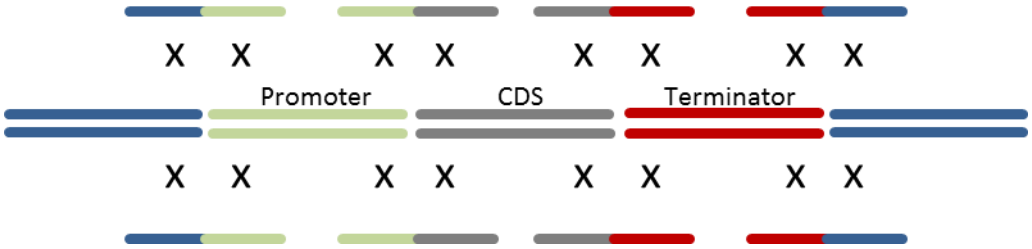


Figure 12: Use of stitching oligonucleotides. For stitching two or more fragments not sharing end homology, the usage of 72 bases long sense and antisense oligonucleotides containing 36 bases of each of the two fragments to be joined is possible.

As a further alternative the PaperClip method (Trubitsyna et al., 2014) can be employed. This method is similar to the one using stitching oligonucleotides, but makes use of reusable 'clips' and therefor helps to reduce primer expenses. For more details see:

> Trubitsyna, M., et al. (2014). "PaperClip: rapid multi-part DNA assembly from existing libraries." Nucleic Acids Res 42(20): e154.

Producing inserts by digestion:

To assemble Level 1 modules from individual Level 0 transcriptional units, or Level 2 modules from several Level 1 modules, it is possible and recommended to release the assembly parts from the respective vector backbones by digestion with restriction enzymes (8-mer cutters or I-*Sce*I, respectively).

- Note, that short non-homologues overlaps resulting from restriction enzyme cleavage are tolerated and removed by SLiCE, TAR and NEBuilder HiFi-DNA assembly reactions, but not by standard Gibson assembly.
- It is necessary to gel-purify the desired products before use in any assembly reaction.

Producing vector backbones by digestion:

Vector backbones need to be digested before use in assembly reactions. Level 0 vectors possess a *ccdB* cassette to avoid empty backbone background in this step (**Figure 6**). Therefore, purification of the *Hin*dIII digested and dephosphorylated vector is not necessary. Level 1 and Level 2 backbones confer ampicillin resistance, while Level 0 vectors carry a kanamycin resistance marker. This avoids background from Level 0 vectors in Level 1 assemblies. As we recommend performing Level 2 assemblies by *in vivo* TAR no marker change is necessary from Level 1 to Level 2. Level 1 and Level 2 backbones need to be digested with *Pac*I prior to TAR assembly reactions. Dephosphorylation and gel purification are recommended, if SliCE or HiFi DNA assembly is used.

4.2. Workflow

Creating a new xYAC always follows the same workflow. The first step is to assemble individual (transcriptional) units in separate Level 0 vectors. To this end, five different sets of Level 0 vectors (A - E) are provided. For each transcriptional unit in the later xYAC, the assembly of one Level 0 construct is needed. The choice of the Level 0 vector backbone determines the position of each unit in the xYAC (see Section 3).

In the second step, each Level 0 setconsisting of up to five Level 0 modules, is combined in corresponding Level 1 vectors to create Level 1 modules. Each set of Level 0 vectors corresponds to one specific Level 1 vector (Level 0 set A corresponds to Level 1 vector pL1A, Level 0 set B corresponds to Level 1 vector pL1B, and so on). If only one Level 1 vector is needed (for assemblies consisting of up to five transcriptional units), this vector can be used as the destination vector for expression in *S. cerevisiae* as Level 1 and Level 2 vector backbones just differ in their homology regions.

The individual Level 1 vectors can finally be assembled into one Level 2 assembly unit. This final vector is a xYAC, suitable for expression in *S. cerevisiae*. Other host organisms can be used when the assembly product is subcloned into a compatible vector (see Section 4.3.4).

4.3. Individual protocols for xYAC assembly

Before starting an experiment, carefully plan the final construct according to the guidelines outlined in this manual (Section 3) and choose the right combinations of vectors/homology regions provided with the toolbox.

4.3.1. Level 0 assembly – Assembly of transcriptional units

Generation of Level 0 constructs is the first step in the modular assembly strategy. These constructs usually contain transcriptional units composed of a promoter, a CDS and a terminator, but can also consist of any other combination of DNA fragments (**Figure 13**). Each transcriptional unit is flanked by two homology regions, which originate from the Level 0 backbone. The identity of the homology region, thus the choice of the Level 0 backbone, determines the later position of the transcriptional unit in the final construct.

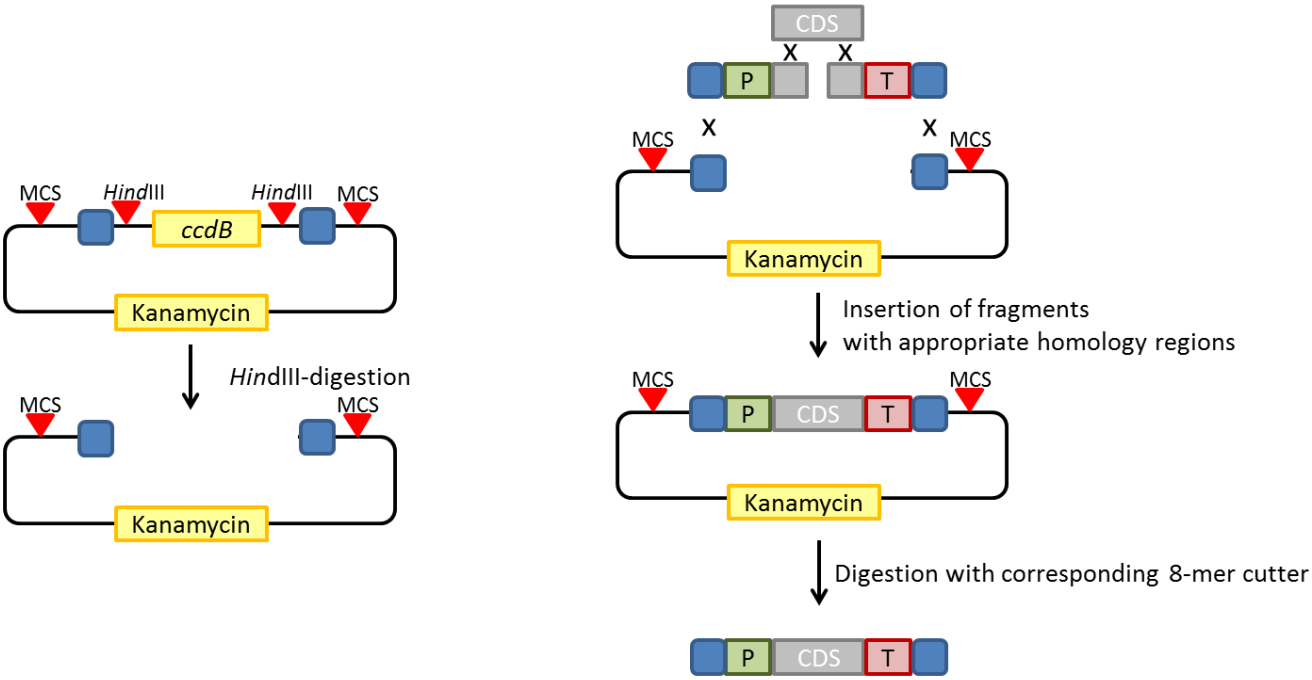


Figure 13: Level 0 assembly. *Hin*dIII-digestion of the Level 0 vector is necessary for insertion of PCR-amplified parts with homology to the vector backbone or to each other, respectively, to build up a Level 0 vector containing a full transcriptional unit. The Level 0 unit can later be released by digestion with one of the 8-mer cutting restriction enzymes cleaving in the MCS.

Level 0 assembly:

1) Digest the Level 0 vector of choice using *Hin*dIII. Heat-inactivate the enzyme according to the manufacturer´s protocol.

2) Add the required homology regions to all the parts of the transcriptional unit (**Figure 13**). This is usually done by PCR amplification according to Section 4.1. Gel- or column-purify the PCR products, if necessary.

Notes:

- *In a four-fragment assembly (e.g. vector, promoter, CDS, and terminator) it is sufficient to add homology regions to the promoter and terminator (**Figure 13**).*
- *To clone several Level 0 constructs containing the same promoter/terminator combinations, but different CDS, it might be advantageous to assemble a Level 0 construct with only promoter and terminator, separated by a sequence containing a suitable restriction site for further insertion of the CDS. Integration of many different DNA fragments by ligation without having to change the homology regions is straightforward. A further approach is a combinatorial assembly, in which the promoter and terminator contain homology to the vector, but not to the CDS, while the different CDS contain homologies to promoter and terminator (**Figure 14**).*

3) Set up an assembly reaction with the *Hin*dIII-linearized Level 0 vector and all necessary fragments according to Section 5. The Level 0 vectors included in the kit are suitable for SLiCE, NEBuilder HiFi-DNA assembly, and TAR. The *E. coli* selection marker is kanamycin resistance, the yeast auxotrophic marker gene is *URA3*.

4) Verify clones by (yeast) colony PCR, isolate plasmids and sequence the assembly junctions.

Note: Protocols for yeast colony PCR and rescue of plasmids from yeast are provided in Section 5.3.

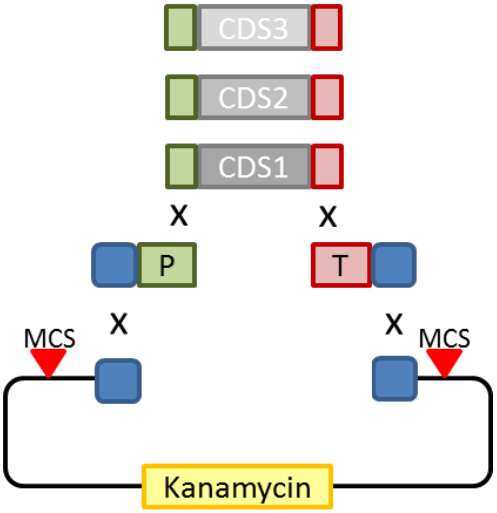


Figure 14: Cloning of different coding sequences with the same promoter and terminator sequences into a Level 0 vector. For combinatorial assembly of different coding regions with the same promoter and terminator sequences, the regions homologous to promoter and terminator can be added to the respective CDS, while promoter and terminator each have only one added sequence homologous to the vector parts.

4.3.2. Level 1 assembly – assembly of up to five Level 0 cassettes into one Level 1 vector

Up to five Level 0 constructs can be assembled into the appropriate Level 1 vector. The left homology region of the first Level 0 vector of one series (e.g. A0 in the Level 0 A set) fits the left homology region of the Level 1 vector (pL1A) and all following Level 0 constructs overlap to each other by their first and last 36 bp. The right homology region of the last Level 0 vector (e.g. AR) fits the right homology region of the Level 1 vector (**Figure 15**).

Notes:

- *When less than five Level 0 vectors are used for one Level 1 assembly, it is important to choose the correct Level 0 vector which provides the homology to Level 1 (***Table 2, Figure 16***).*
- *Level 1 vectors are yeast shuttle vectors and are provided with a 2µ-origin (high copy) or a CEN/ARS origin (low copy), respectively. If a Level 1 vector is the destination*

vector for the assembly and S. cerevisiae is the host for gene expression, choose which Level 1 vector fits best. Otherwise, we suggest using the CEN/ARS-origin. The assembled fragments can also be transferred to an integrative yeast vector (Section 3.3).

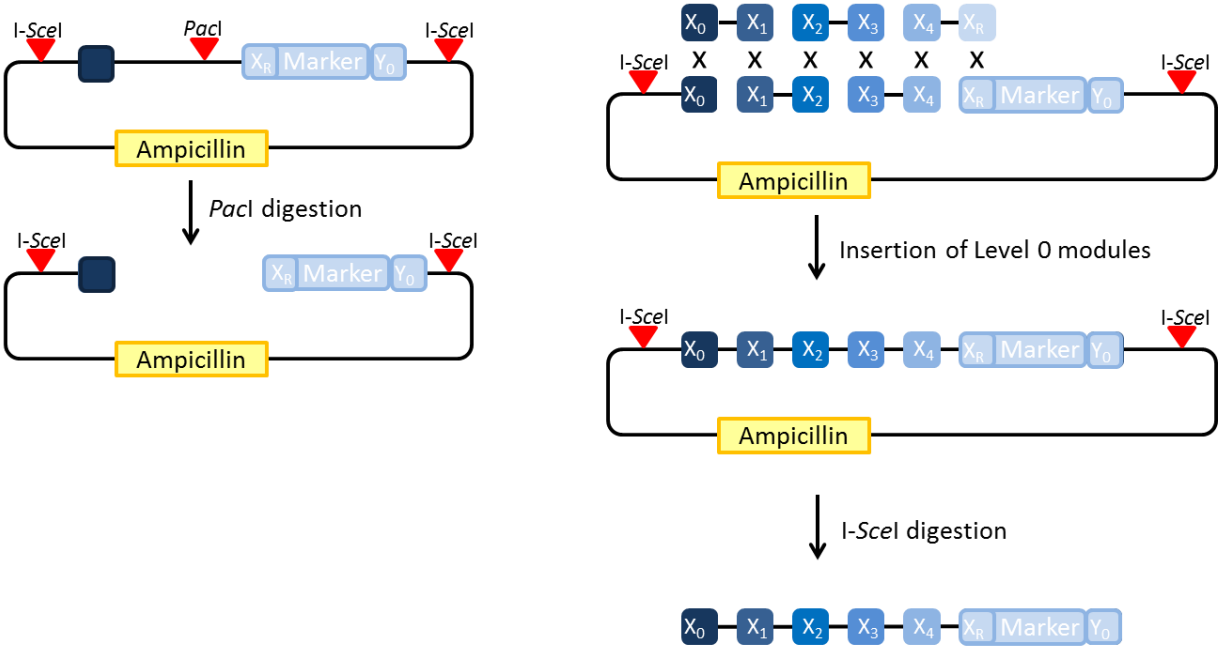


Figure 15: Level 1 assembly with five Level 0 constructs. Five Level 0 fragments can be combined via homology regions into a *Pac*I-digested Level 1 vector.

Assembly of a Level 1 module:

1) Linearize the required Level 1 vector(s) by *Pac*I-digestion. Heat-inactivate the enzyme.

   Note: Overnight digestion, dephosphorylation and subsequent gel purification are recommended to reduce background of uncut vector in subsequent in vitro assembly reactions.

2) Digest Level 0 constructs with a suitable 8-mer cutter. The Level 0 backbones provide sites for *Asc*I, *Sbf*I, *Swa*I, *Fse*I and *Pme*I on both sites of the transcriptional unit. Gel-purify the digestion fragment representing the assembly piece. Alternatively, it is possible to PCR-amplify the assembly piece, including the homology regions.

3) Set up an assembly reaction with the *Pac*I-linearized Level 1 vector and all Level 0 assembly pieces. The Level 1 vector is suitable for SLiCE, NEBuilder HiFi-DNA assembly, and TAR. The bacterial selection marker is ampicillin resistance. Depending on the Level 1 vector used, the yeast selection marker varies (**Table 2**). In yeast, auxotrophic selection for the Level 1 vector only works for assembled Level 1 modules. The empty Level 1 vector cannot be selected for in yeast. This is due to the nature of the homology between the last Level 0 vector and the Level 1 vector.

Notes:

- *For Level 1 assemblies with three or more inserts we strongly recommend using NEBuilder HiFi-DNA assembly or TAR, rather than SLiCE.*
- *If internal homologous sequences occur in the desired Level 1 module, for example if the same signal peptide is employed in all five Level 0 constructs, a careful analysis of positive clones from a TAR reaction for sequence integrity is necessary. We have observed a tendency towards undesired recombination between such sequences during TAR. In these cases we recommend to use NEBuilder HiFi-DNA assembly instead.*

4) Verify clones by (yeast) colony PCR, isolate plasmids, perform analytic digestions and sequence the assembly junctions.

Note: Protocols for yeast colony PCR and rescue of plasmids from yeast are provided in Section 5.3.

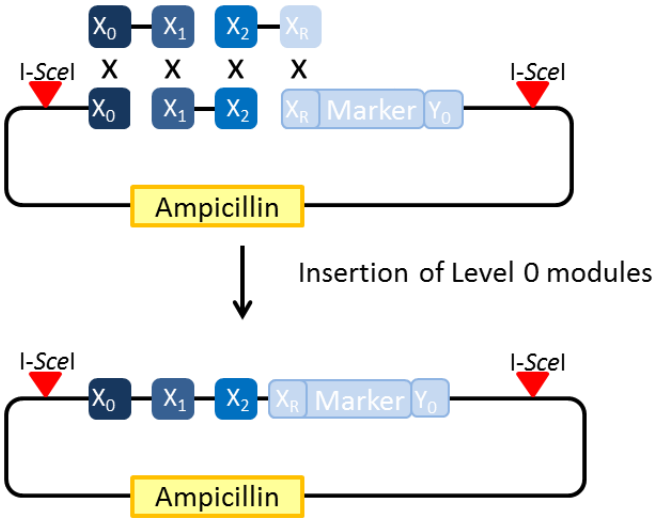


Figure 16:Level 1 assembly with three Level 0 constructs. Combination of three Level 0 units into a *Pac*I-digested Level 1 vector. The homology region of the last Level 0 fragment needs to fit with the right homology region of Level 1 vector. The last Level 0 fragment completes the yeast marker, which is promoter-less in the empty Level 1 backbone, thereby allowing for selection of yeast colonies containing correct insertions in the Level 1 vector on appropriate dropout media plates.

4.3.3. Level 2 assembly – assembly of Level 1 modules in a single Level 2 vector

Up to five Level 1 modules can be combined in an appropriate Level 2 vector. Level 2 vectors for two, three, four and five Level 1 modules are provided as high- or low-copy versions for *S. cerevisiae* (**Table 3**). Choose the desired vector type based on the experimental needs (**Figure 17**).

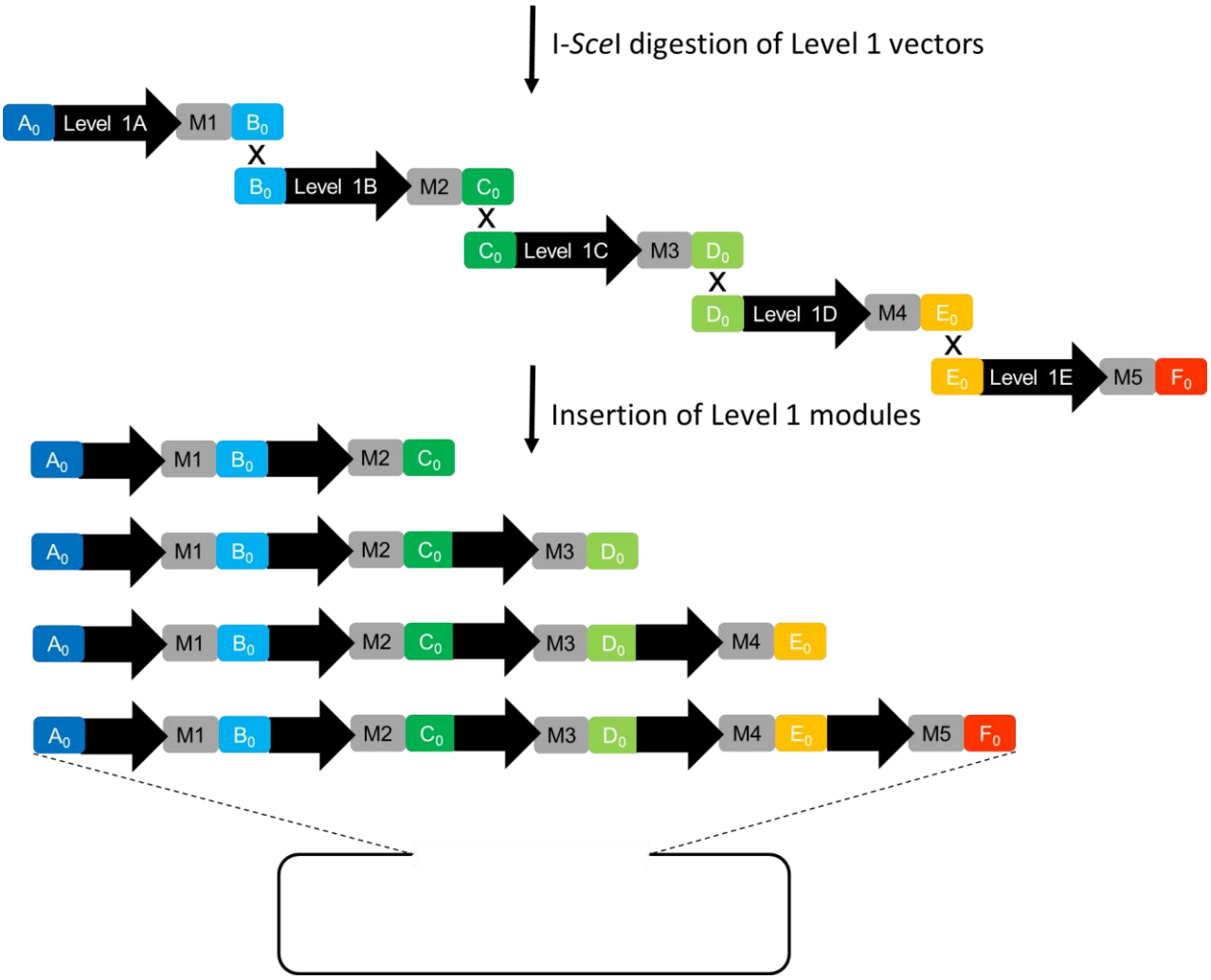


Figure 17: Level 2 assembly. Assembly of five I-*Sce*I-digested Level 1 modules into a *Pac*I-digested Level 2 vector.

1. Digest the required Level 2 vector with *Pac*I. Heat-inactivate the enzyme.

   *Note: Overnight digestion, and subsequent gel purification are usually **not** necessary for the following in vivo assembly.*

2. Digest the required Level 1 constructs with I-*Sce*I to release the assembly units. We recommend the following reaction set up:

   Level 1 vector 2 µg

   10x CutSmart buffer (NEB) 5 µl

   I-SceI (NEB, 5 U/µl) 2 µl

   ddH_2O to 50 µl final volume

   Digest overnight at 37 °C, inactivate at 65 °C for 20 min, and gel-purify the desired module.

3. Combine all necessary parts in a TAR reaction and select positive yeast clones on appropriate dropout medium. See **Table 3** Table 3for selection of the correct dropout combination.

4. Verify clones by (yeast) colony PCR, isolate plasmids, perform analytic digestions, and sequence the assembly junctions.

   Note: Protocols for yeast colony PCR and rescue of plasmids from yeast are provided in Section 5.3.

After successful assembly, the whole construct can be transferred to different integrative yeast vectors or any compatible vector for the destination organism (see Section 4.3.4).

4.3.4. Transferring the multi-gene assembly to different destination vectors

If you wish to express your final Level 1 or Level 2 assembly in a host different from *E. coli* or yeast, please follow one of the following three options. For all three options, it is necessary to linearize the expression vector for the intended host. This may be done by restriction digestion or preferably by PCR amplification. Once the vector is linearized, one of the following three options may be used (**Figure 18**).

Option 1: Convert your expression vector into an AssemblX compatible Level 2 vector

- For this option, make sure that your expression vector does not contain a *Pac*I site.
- Design primers that amplify the yeast replication origin and the appropriate pair of homology regions from one of the available AssemblX Level 2 vectors. As a template, choose the same Level 2 vector you would have used if yeast was your final host for expression. If you are not sure, take a look at the "*Level 2*" section in the AssemblX protocol that you received after submitting your assembly to the web tool.
- Equip primers with overlaps compatible to your linearized expression vector and use these primers to amplify the cassette described above.
- In an appropriate assembly reaction (e.g. SLiCE or Gibson assembly; do **not** use TAR as your vector does not yet contain a selection marker), combine the following:
    - Linearized expression vector
    - PCR-amplified vector conversion cassette
- Identify positive clones by colony PCR and restriction analysis.
- Use the modified expression vector instead of the designated AssemblX Level 2 vector for the final TAR mediated Level 2 assembly.
- Proceed with construct verification and isolation. Transform verified constructs into your final host.

Option 2: Subclone AssemblX Level 2 module into your expression vector via recombination.

- For this option, make sure that your expression vector does not contain a *Pac*I site.
- Design primers that amplify the appropriate pair of homology regions from one of the available AssemblX Level 2 vectors. As a template, choose the same Level 2 vector you would have used if yeast was your final host for expression. If you are not sure, take a look at the "*Level 2*" section in the AssemblX protocol that you received after submitting your assembly to the web tool.
- Equip primers with overlaps compatible to your linearized expression vector and use these primers to amplify the cassette described above.
- In an appropriate assembly reaction (e.g. HiFi DNA assembly or SLiCE) combine the following:
    - Linearized expression vector
    - PCR-amplified recombination cassette
- Identify positive clones by colony PCR and restriction analysis.
- Use the modified expression vector to subclone your AssemblX Level 2 construct. For this, digest your verified Level 2 construct with I-*Sce*I, purify the fragment and perform *in-vitro* recombination (e.g. HiFi DNA assembly, SLiCE) with *Pac*I digested modified expression vector.
- Identify positive clones by colony PCR and restriction analysis and proceed with transformation in your final host.

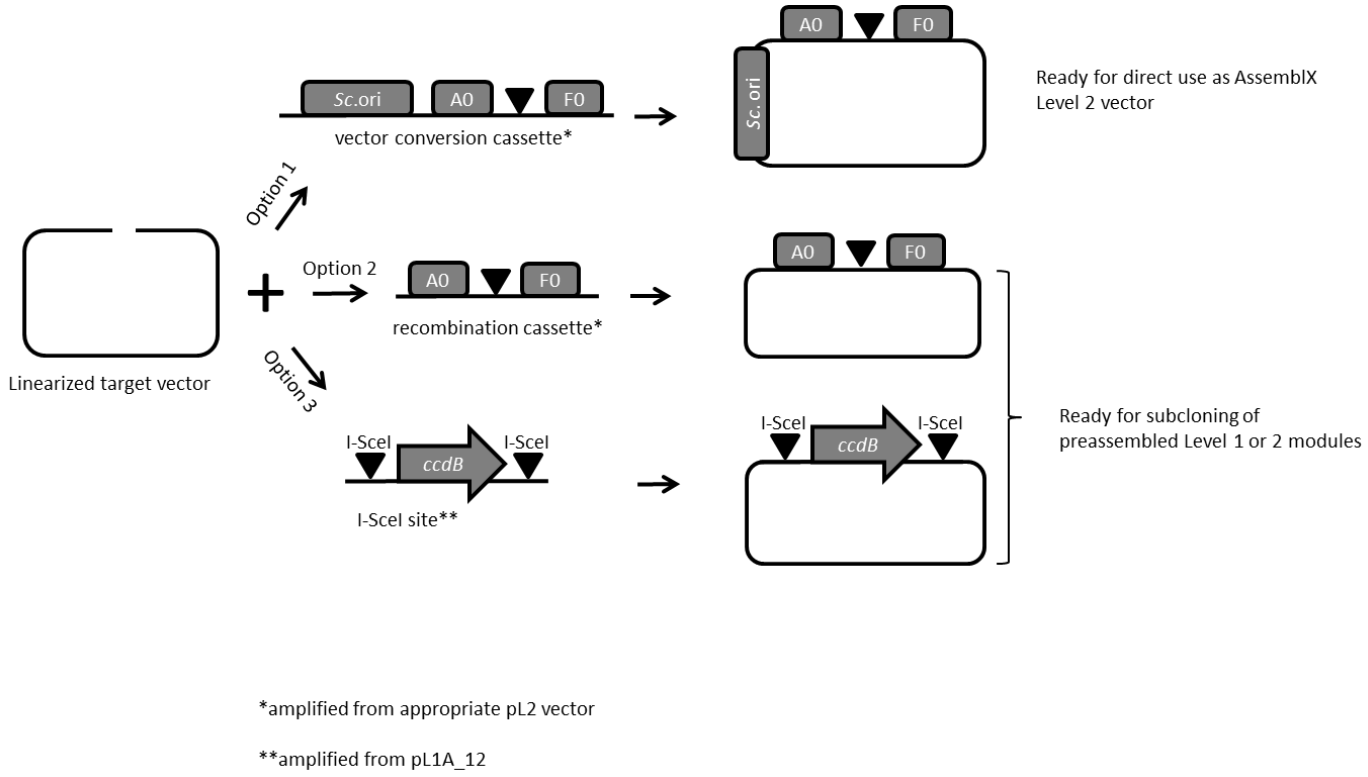


Figure 18: Three different possibilities to convert any expression vector into an AssemblX compatible vector. Option 1: Equip expression vector with a yeast replication origin and use the converted vector directly in a TAR mediated Level 2 Assembly. The appropriate cassette can be amplified from the existing AssemblX Level 2 vectors. Option 2: Equip expression vector with a recombination cassette compatible to your Level 2 construct. The appropriate recombination cassette can be amplified from the existing AssemblX Level 2 vectors. Following the vector conversion, the modified vector can be used to subclone an I-SceI released Level 2 module via overlap based cloning. Option 3: Equip expression vector with an I-SceI-ccdB-I-SceI cassette amplified from vector pL1A_12. The modified vector can be digested with I-SceI and ligated to any I-SceI released fragment from any AssemblX Level 1 or 2 construct.

Option 3: Subclone AssemblX Level 1 or 2 module into your expression vector via restriction & ligation.

- For this option, make sure that your expression vector does not contain an I-*Sce*I site.
- Design primers that amplify the I-*Sce*I - *ccdB* - I-*Sce*I cassette from pL1A_12.
- Equip primers with overlaps compatible to your linearized expression vector and use these primers to amplify the cassette described above.
- In an appropriate assembly reaction (e.g. HiFi DNA assembly or SLiCE) combine the following:
    - Linearized expression vector
    - PCR-amplified I-*Sce*I - *ccdB* - I-*Sce*I cassette

    For this step, make sure to use ccdB survival cells

- Identify positive clones by colony PCR and restriction analysis.
- Use the modified expression vector to subclone your AssemblX Level 1 or 2 construct. For this, digest your verified construct with I-*Sce*I, purify the fragment and perform a ligation with your I-*Sce*I digested modified expression vector.
- Identify positive clones by colony PCR and restriction analysis and proceed with transformation in your final host.

5. Technical advice

This kit relies on established recombination-based cloning methods which avoid the insertion of scar sequences in the assemblies and is independent from restriction enzymes. Recombinational cloning is based on homology regions present on the fragments to be assembled. The cloning methods used here are SliCE, NEBuilder HiFi DNA assembly (a modified Gibson assembly), and TAR cloning. Suggestions for setting up assembly reactions using these different methods are given below. However, for more details we recommend the following publications and handbooks:

SLiCE:

- Zhang, Y., Werling, U., and Edelmann, W. (2012) SLiCE: a novel bacterial cell extract-based DNA cloning method. Nucleic Acids Research *40*, e55.

Gibson assembly/NEBuilder HiFi DNA assembly:

- Gibson, D.G. (2011). Enzymatic Assembly of Overlapping DNA Fragments. In Methods in Enzymology, V. Christopher, ed. (Academic Press), pp. 349-361.

- NEB manual (https://www.neb.com/~/media/Catalog/All-Products/709D232D72C045D2B2B1089A89DC879F/Datacards%20or%20Manuals/manualE2621.pdf)

TAR:

- Gibson, D.G. (2009). Synthesis of DNA fragments in yeast by one-step assembly of overlapping oligonucleotides. Nucleic Acids Research *37*, 6984-6990.

- Gibson, D.G., Benders, G.A., Axelrod, K.C., Zaveri, J., Algire, M.A., Moodie, M., Montague, M.G., Venter, J.C., Smith, H.O., and Hutchison, C.A. (2008). One-step assembly in yeast of 25 overlapping DNA fragments to form a complete synthetic Mycoplasma genitalium genome. Proceedings of the National Academy of Sciences *105*, 20404-20409.

- Larionov, V., Kouprina, N., Graves, J., Chen, X.N., Korenberg, J.R., and Resnick, M.A. (1996). Specific cloning of human DNA as yeast artificial chromosomes by

transformation-associated recombination. Proceedings of the National Academy of Sciences of the United States of America *93*, 491-496.

5.1. Setting up a SLiCE reaction

Prepare a bacterial lysate and 10x SLiCE buffer following the published protocol (Zhang et al., 2012). We recommend to store the lysate in small aliquots at -80°C, as repeated freeze and thaw cycles strongly reduce cloning efficiency. **Table 5** shows a standard reaction setup. Following incubation we usually transform 2 µl of the assembled product into highly competent *E. coli* cells. Cloning efficiency is usually sufficient for the assembly of up to three fragments. The SLiCE reaction tolerates non-homologous overhangs, e.g. overhangs created by digestion with restriction enzymes.

Table 5: Setup of a SLiCE reaction.

Total amount of fragments	1-3
Backbone	50-100 ng
Recommended molar ratio backbone:insert	1:10
SLiCE lysate	1 µl
SLiCE buffer	1 µl
Total volume	10 µl
Incubation time	60 min
Incubation temperature	37°C

5.2. Setting up a NEBuilder HiFi DNA assembly reaction

The commercially available NEBuilder HiFi DNA assembly master mix is a highly efficient version of the Gibson assembly and we suggest to use this method for assembly of up to five fragments. In our hands the reaction setup shown in **Table 6** gives the best results. The final reaction volume can be scaled down to 5 µl for simple assemblies. Usually, 2 µl assembly product are transformed into highly competent *E. coli* cells. For more details see the manual provided by NEB.

Table 6: Setup for a NEBuilder HiFi DNA assembly reaction.

	Amount of fragments used for assembly	
	2-3 fragment assembly	4-6 fragment assembly
Backbone	50-100 ng	50-100 ng
Recommended molar ratio backbone:insert	1:2	1:1
Gibson Assembly Master Mix (2x)	10 µl	10 µl
Total volume	20 µl	20 µl
Incubation time	15 min	60 min
Incubation temperature	50°C	50°C

5.3. Setting up a TAR reaction and analyzing yeast clones

TAR is an *in vivo* assembly method in *S. cerevisiae*. Basically, the yeast is transformed with linear DNA fragments, sharing end homologies. The yeast DNA repair system then assembles the fragments into a single circular plasmid, which can be selected using marker genes. Any yeast transformation technique, which yields sufficient transformation efficiency, is suitable for a TAR reaction. We usually use a LiAc/PEG method (Gietz and Schiestl, 2007) and transform 100 ng of each fragment (**Table 7**).

Potentially positive yeast clones are usually first analyzed by yeast colony PCR. Various protocols exist. We recommend to resuspend a small amount of yeast cells in 50 µl 0.02 M NaOH, incubate at 37°C for 10 min and use 1 µl thereof as template for a 10-µl PCR reaction with your favorite PCR reagents.

For sequencing analysis of positive clones, plasmid rescue from yeast and subsequent transformation in *E. coli* is necessary. In our hands, the *Zymoprep Yeast Plasmid Miniprep II* Kit (Zymo Research, #D2004) yields sufficient amounts of plasmid for retransformation in highly competent *E. coli* cells, even if the plasmid is large. For the latter, it is advantageous to double the amounts of cells and reagents used to lyse the cells and to extend lysis to 60 min. Furthermore, using pre-warmed (70°C) TE buffer for elution is recommended.

Table 7: Set up of a TAR reaction.

Total amount of fragments	2-10 fragment assembly
Backbone	100 ng
Inserts	each 100 ng

6. References

Gibson, D.G. (2011). Enzymatic Assembly of Overlapping DNA Fragments. In Methods in Enzymology, V. Christopher, ed. (Academic Press), pp. 349-361.

Gietz, R.D., and Schiestl, R.H. (2007). High-efficiency yeast transformation using the LiAc/SS carrier DNA/PEG method. Nature Protocols *2*, 31-34.

Hillson, N.J., Rosengarten, R.D., and Keasling, J.D. (2011). j5 DNA Assembly Design Automation Software. ACS Synthetic Biology *1*, 14-21.

Larionov, V., Kouprina, N., Graves, J., Chen, X.N., Korenberg, J.R., and Resnick, M.A. (1996). Specific cloning of human DNA as yeast artificial chromosomes by transformation-associated recombination. Proceedings of the National Academy of Sciences of the United States of America *93*, 491-496.

Trubitsyna, M., Michlewski, G., Cai, Y., Elfick, A., and French, C.E. (2014). PaperClip: rapid multi-part DNA assembly from existing libraries. Nucleic Acids Research *42*, e154.

Zhang, Y., Werling, U., and Edelmann, W. (2012). SLiCE: a novel bacterial cell extract-based DNA cloning method. Nucleic Acids Research *40*, e55.